NIST SPECIAL PUBLICATION 400-81

U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology

Semiconductor Measurement Technology:

Automatic Determination of the Interstitial Oxygen Content of Silicon Wafers Polished on Both Sides

Warren K. Gladden, Stephen R. Slaughter,
Walter M. Duncan, and Aslan Maghdadi

Semiconductor Measurement Technology:

Automatic Determination of the Interstitial Oxygen Content of Silicon Wafers Polished on Both Sides

Warren K. Gladden,* Stephen R. Slaughter,†
Walter M. Duncan,† and Aslan Baghdadi,*

* Semiconductor Electronics Division
Center for Electronics and Electrical Engineering
National Engineering Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899

†Texas Instruments, Incorporated
Dallas, TX 75265

November 1988

NOTE: As of 23 August 1988, the National Bureau of Standards (NBS) became the National Institute of Standards and Technology (NIST) when President Reagan signed into law the Omnibus Trade and Competitiveness Act.

U.S. DEPARTMENT OF COMMERCE, C. William Verity, Secretary
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, Ernest Ambler, Director
(formerly National Bureau of Standards)

Library of Congress Catalog Card Number: 88–600601
National Institute of Standards and Technology Special Publication 400-81
Natl. Inst. Stand. Technol. Spec. Publ. 400-81, 71 pages (Nov. 1988)
CODEN: NSPUE2

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1988

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402–9325

TABLE OF CONTENTS

LIST OF FIGURES

Semiconductor Measurement Technology:
Automatic Determination of the Interstitial Oxygen Content
of Silicon Wafers Polished on Both Sides

Warren K. Gladden,*‡ Stephen R. Slaughter,†
Walter M. Duncan,† and Aslan Baghdadi*

*National Institute of Standards and Technology
(formerly National Bureau of Standards)
Gaithersburg, MD 20899

†Texas Instruments, Incorporated
Dallas, TX 75265

ABSTRACT

This Special Publication contains FORTRAN and PASCAL computer programs which implement an ASTM test method for the automatic determination of the interstitial oxygen content of silicon. The programs are to be used as illustrative examples by programmers wishing to implement the ASTM algorithm on their computers. The Publication also includes sample data that can be used to test the computer programs. The sample data are included in two forms: in print, and on an MS-DOS floppy disk.

Key words: algorithm; analysis of infrared spectra; computer programs; interstitial oxygen; silicon.

I. INTRODUCTION

This work is an outgrowth of an American Society for Testing and Materials (ASTM) project to automate test methods for the determination of the oxygen and carbon content in silicon using IR absorption. The ASTM task force on this project recommended a test procedure for obtaining the IR spectra and an algorithm for computing the oxygen content from the spectra. This test procedure has been proposed for adoption by ASTM Committee F-1 as a Standard Test Method. This Special Publication includes as examples a FORTRAN computer program and a PASCAL program, each of which implements the ASTM algorithm. The FORTRAN program was written at the National Institute of Standards and Technology, and the PASCAL progran was written at Texas Instruments. Also included are example spectra that can be used to test any program written implementing the ASTM algorithm.

‡ Now at AT&T Bell Laboratories, Allentown, Pennsylvania.

The proposed ASTM method can be used to analyze IR spectra obtained on either dispersive (DIR) or Fourier-transform (FT-IR) infrared spectrophotometers. It can be used with samples ranging in thickness from 0.3 mm to 2.5 mm. The samples must be polished on both sides. The lower resistivity limits are 0.05 Ω-cm for n-type and 0.5 Ω-cm for p-type samples. There are no upper resistivity limits. The oxygen content of float-zoned silicon has been measured with resistivities well above 100 Ω-cm. The range of oxygen concentrations measured by this method is 1×10^{16} to 2×10^{18} atoms/cm^{-3}.

II. OUTLINE OF THE ALGORITHM

The interstitial oxygen content of silicon can be determined by measuring the net peak height of the oxygen vibrational absorption at 1107 cm^{-1}.§ The transmittance through the sample, taking into account the additional intensity due to multiple reflections between the parallel polished surfaces of the sample, is given by¶

$$T = \frac{(1-R)^2 e^{-\alpha d}}{1 - R^2 e^{-2\alpha d}}, \tag{1}$$

and rearrangement of this equation yields the absorption coefficient as

$$\alpha = -\frac{1}{d} \ln\left\{ \frac{-0.49 + \sqrt{0.2401 + 0.36T^2}}{0.18T} \right\}, \tag{2}$$

where the silicon reflectivity, R, has been set equal to 0.30, its value over the carrier concentrations and frequencies considered here. Figure 1 shows a typical silicon transmittance spectrum and figure 2 shows the region about the oxygen peak where the transmittance data have been converted to absorption coefficient using eq (2). To determine this peak height, a baseline is drawn between the two minima, one on either side of the peak. The peak height is determined at 1107 cm^{-1}, and the baseline absorption is determined by finding the intercept of the baseline at 1107 cm^{-1}. The net peak height is then the difference between these two values of absorption coefficient. A calibration constant has been determined relating this net peak height to the quantity of interstitial oxygen in parts per million (atomic) using absolute techniques such as charge particle and/or photon activation

§ See ASTM Test Method F 121, 1987 Annual Book of ASTM Standards, V. 10.05, p. 247.

¶ See ASTM Standard Practices F 120, 1987 Annual Book of Standards, V. 10.05, p. 245.

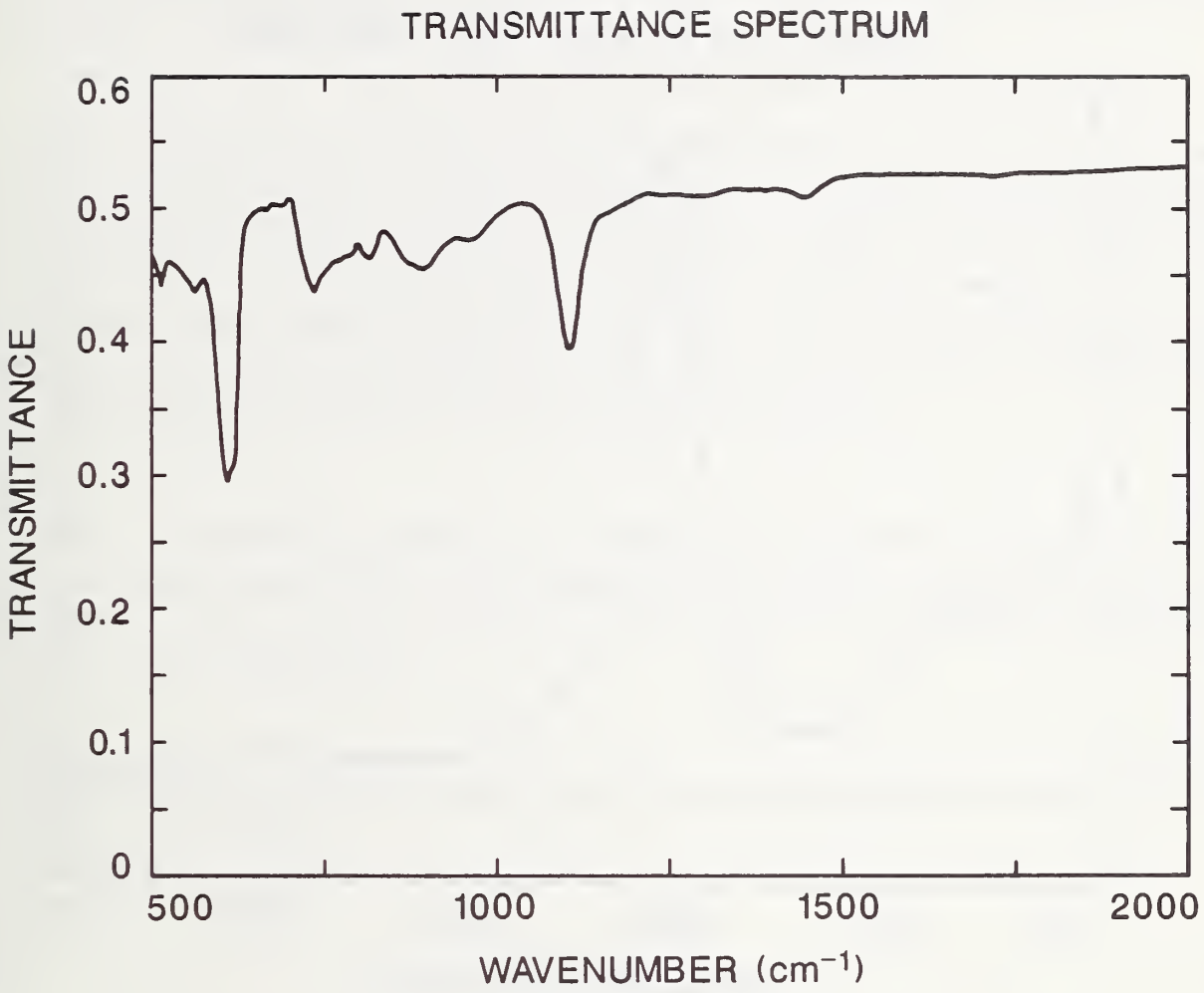

Figure 1: Transmittance spectrum of a p-type Czochralski silicon wafer with a resistivity of 9 Ω-cm and an oxygen content of 20 parts per million (atomic).

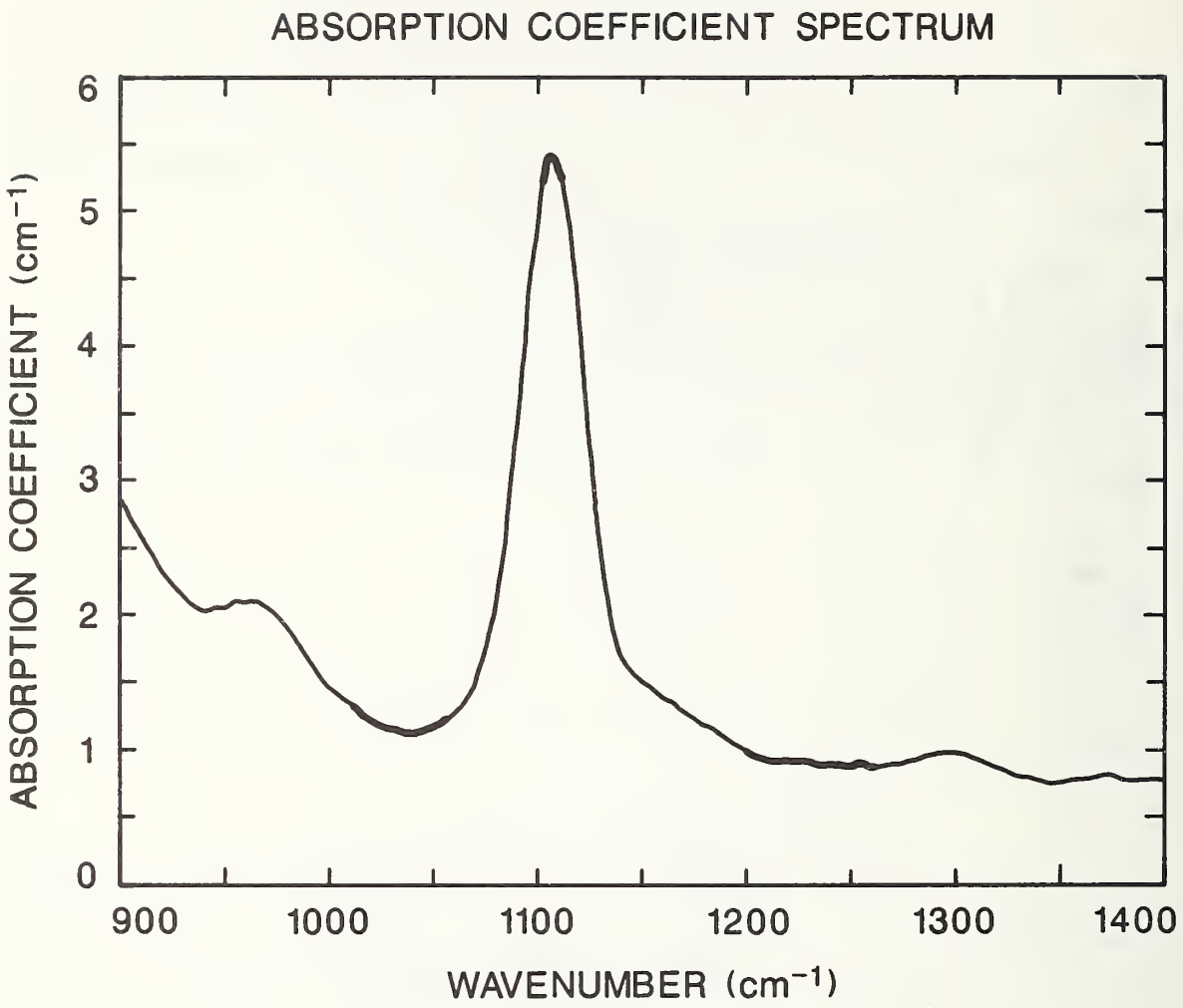

Figure 2: Absorption coefficient spectrum of a p-type Czochralski silicon wafer with a resistivity of 9 Ω-cm and an oxygen content of 20 parts per million (atomic). The thick sections of the line indicate the wavenumber regions used to calculate the oxygen content (see page 2).

analysis. This constant is 6.28 cm·ppma.[+]

This procedure was converted to a computer algorithm. The following is an outline of the algorithm.

- The measured IR transmittance of the sample is read from a data file and only those transmittances between 1000 and 1300 wavenumbers are stored. Three regions about the interstitial oxygen minimum, 1010 to 1060 cm^{-1}, 1200 to 1260 cm^{-1}, and 1090 to 1123 cm^{-1}, are considered for further analysis. The first two regions are used to determine the baseline for the minimum. By the method of least-squares, a third-order polynomial is fitted to the transmittance data in each of these two regions. For the absorption maximum, a fourth-order polynomial is used.

- The two transmittance maxima, the minimum, and the wavenumbers at which they occur are determined from the least-squares-fit curves. The transmittance values are denoted by T_{max1}, T_{peak}, and T_{max2}. The respective wavenumbers are denoted by σ_{max1}, σ_{peak}, and σ_{max2}.

- The baseline transmittance is determined from a linear interpolation between the two maxima of the form

$$T_{base}(\sigma) = T_1 + \frac{T_2 - T_1}{\sigma_2 - \sigma_1}(\sigma - \sigma_1).$$

- α_{peak} and α_{base} are determined by solving eq (2) using the appropriate calculated transmittance.

- The net absorption coefficient is calculated by

$$\alpha_{net} = \alpha_{peak} - \alpha_{base}.$$

[+] A. Baghdadi, W. M. Bullis, M. C. Croarkin, Li Yue-zhen, R. I. Scace, R. W. Series, P. Stallhofer, and M. Watanabe, to be published in the Journal of the Electrochemical Society.

Lattice absorption at 1107 cm^{-1} due to lattice bands contributes an additional net absorption of 0.5 cm^{-1},# so the oxygen content is calculated by

$$\text{ppm} \quad \text{atomic} = 6.28 \times (\alpha_{net} - 0.5).$$

This algorithm has been implemented in FORTRAN and PASCAL computer programs, which are listed in the sections IIIA and IIIB, respectively. These programs should only be considered as examples rather than as the approved methods for implementing the algorithm. Section IV contains four sample transmission spectra. These spectra (plus two more) have also been written on $5\frac{1}{4}$–in. MS-DOS◇ floppy disks that can be found in a pocket on the end of this Special Publication. Section V is a listing of the output from the FORTRAN and PASCAL programs listed in section III, as applied to all six sample spectra. The thicknesses of the silicon wafers used to produce the IR spectra are included in the output listed in section V. The oxygen contents determined for these spectra should match the oxygen contents listed in this output. Thus, these spectra can be used to test a computer program based upon the algorithm by comparing the oxygen content computed using the program to the listed oxygen content. The oxygen contents in section V are listed to four significant figures, rather than the three significant figures which is appropriate considering the precision of the IR measurement, in order to facilitate the comparison of computer programs.

See ASTM Test Method F 121, 1988 Annual Book of ASTM Standards, V. 10.05, to be published.

◇ Certain commercial equipment, instruments or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

ANNOTATED PROGRAMS

Section IIIA. FORTRAN Program

```
      PROGRAM ASTMDSP
C  ************************** ASTMDSP **********************************
C  *                                                                   *
C  * THIS PROGRAM COMPUTES THE INTERSTITIAL OXYGEN CONTENT OF           *
C  * DOUBLE-SIDE POLISHED SILICON SAMPLES IN PARTS PER MILLION ATOMIC.  *
C  * THE PROCEDURE IS A MODIFICATION OF THAT PRESENTED IN THE GREY      *
C  * PAGES OF THE ASTM HANDBOOK. INSTEAD OF TAKING AVERAGE VALUES FOR   *
C  * THE TWO MAXIMA AND THE MINIMUM, A LEAST-SQUARES POLYNOMIAL IS FIT  *
C  * TO SEVERAL DATA POINTS AROUND EACH FEATURE. ALSO, THE WAVENUMBERS  *
C  * WHERE THE MAXIMA AND MINIMUM OCCUR ARE DETERMINED FROM THE FIT. A  *
C  * 3RD-ORDER POLYNOMIAL IS USED FOR THE MAXIMA AND A 4TH-ORDER       *
C  * POLYNOMIAL IS USED FOR THE MINIMUM. THE 4TH ORDER FIT WAS CHOSEN   *
C  * IN ORDER TO BE COMPATIBLE WITH THE PROCEDURE GIVEN IN THE ANNEX    *
C  * FOR THE OXYGEN DETERMINATION IN THIN SLICES.                      *
C  *                                                                   *
C  * ------NOTE:                                                       *
C  *             IT IS COMMON PRACTICE TO SPEAK OF THE INTERSTITIAL     *
C  *             OXYGEN PEAK. THIS IS ONLY TRUE WHEN WORKING IN        *
C  *             ABSORBANCE. ALL DATA REDUCTION IN THIS PROGRAM IS      *
C  *             DONE IN TRANSMITTANCE, THUS WE SPEAK OF THE           *
C  *             INTERSTITIAL OXYGEN MINIMUM.                          *
C  *                                                                   *
C  ************************** ASTMDSP **********************************
      DIMENSION ABSRB(500),FUNC(200),IFLAG(6),SIGMA(500),SPEC(600),W(6),
     +  WAVE(600),WEIGHT(200),WORK(500)
      CHARACTER*40 FILEIN,FILEOUT
      DOUBLE PRECISION ANSWER(200)
      INTEGER DAY,YEAR
      REAL MAX1,MAX2,MINPK,MAXCAL,MINCAL,NET,NETABS,NODE(200)
      DATA IFLAG,W,WEIGHT/6*0,1010.,1060.,1090.,1123.,1200.,1260.,
     +  200*1./
 1001 FORMAT(3I2)
 1002 FORMAT(A40)
 1003 FORMAT(F10.4)
 1011 FORMAT(1H1/' ASTMDSP ------------- ',I2,'/',I2.'/',I2,
     +        35X,'PAGE ',I1)
 1012 FORMAT(1H1/' ASTMDSP ------------- ',I2,'/',I2.'/',I2,
     +        34X,'PAGE ',I2)
 1013 FORMAT(' *****spectrum***** ',A40)
 1014 FORMAT('      thickness is ',f6.4,'cm')
 1015 FORMAT(/'  T(',F6.1,') = ',F6.4,'        residual std. dev.=',e10.
     +3/'  T(',F6.1,') = ',F6.4,'        residual std. dev.=',e10.3/'  T
     +(',F6.1,') = ',F6.4,'        residual std. dev.=',e10.3)
 1016 FORMAT(/'  BASELINE TRANSMITTANCE AT ',F6.1,' WAVENUMBERS=',F7.4)
 1017 FORMAT(' ====>  ALPHA(BASE) = ',F5.3,'    ALPHA(PEAK) = ',F7.5/)
 1018 FORMAT('  NET ABSORBANCE PEAK HEIGHT = ',F8.6/)
 1019 FORMAT(10X,'*******************************'/,10X,'* OXYGEN CONTEN
     +T IS ',F5.2,' PPMA *'/,10X,'*******************************'/)
      NLINES=0
      IPAGE=1
C
C  ***********************************************************************
C  *  FILE 'MODEL.RES' IS USED FOR STORAGE OF THE RESULTS. THE OUTPUT *
C  *  IS PRINTED TO THIS FILE IN SUCH A WAY THAT YOU CAN OBTAIN A      *
C  *  FORMATTED HARD-COPY OF THE FILE AT A LATER TIME.                 *
C  ***********************************************************************
C
```

```
      OPEN(UNIT=30,FILE=' MODEL.RES',STATUS='NEW')
C
C  ************************************************************************
C  *  INPUT DATE, NAME OF TRANSMITTANCE FILE, AND THICKNESS OF WAFER.  *
C  ************************************************************************
C
      WRITE(6,*) '  TODAY"S DATE----'
      READ(5,1001)MONTH,DAY,YEAR
    5 WRITE(6,*) '  NAME OF DSP TRANSMITTANCE DATA FILE.'
      READ(5,1002)FILEIN
      WRITE(6,*) '  THICKNESS OF WAFER.'
      READ(5,1003)TAU
      OPEN(UNIT=7,FILE=FILEIN,STATUS='OLD')
        IF(NLINES.EQ.0.AND.IPAGE.EQ.1) THEN
          WRITE(30,1011)MONTH,DAY,YEAR,IPAGE
          NLINES=NLINES+1
        END IF
      WRITE(30,*)'     '
      WRITE(30,*)'     '
      WRITE(30,1013)FILEIN
      WRITE(30,1014)TAU
C
C  ************************************************************************
C  * READ TRANSMITTANCE DATA AND SELECT THOSE DATA POINTS BETWEEN 1000 *
C  * AND 1300 WAVENUMBERS (INCLUSIVE).                                *
C  * FORMAT OF DATA FILE:                                             *
C  *         (1) FIRST LINE CONTAINS THE NUMBER OF LINES OF DATA      *
C  *         (2) EACH SUBSEQUENT LINE CONTAINS THE COUNTER VALUE      *
C  *             CORRESPONDING TO THE LINE OF DATA, WAVENUMBER, AND   *
C  *             TRANSMITTANCE.                                       *
C  ************************************************************************
C
      K=1
      READ(7,*)NCNT
      DO 10 I=1,NCNT
      READ(7,*,ERR=115)LL,WAVENM,TRANSM
        IF((WAVENM.GE.1000.).AND.(WAVENM.LE.1300.)) THEN
          WAVE(K)=WAVENM
          SPEC(K)=TRANSM
          K=K+1
          GO TO 10
        ELSE IF (WAVENM.GT.1300.) THEN
          GOTO 15
        END IF
   10 CONTINUE
   15 NDP=K-1
C
C  ************************************************************************
C  *       DETERMINE WAVENUMBER REGIONS FOR LEAST-SQUARES FIT OF      *
C  *    DATA AT MAXIMA AND MINIMUM ABOUT 1107 WAVENUMBER ABSORPTION   *
C  ************************************************************************
C
```

```
      MAXDEG=3
      KNT=0
      INDEX=-1
      DO 100 I=1,NDP
      IF(WAVE(I).GE.W(1)) INDEX=2
      IF(WAVE(I).GE.W(3)) INDEX=4
      IF(WAVE(I).GE.W(5)) INDEX=6
      IF(INDEX.LT.0)GOTO 100
C
C     ***************************************************************************
C     * CREATE ARRAYS OF NODES AND FUNCTION VALUES FOR LEAST SQUARES FIT. *
C     ***************************************************************************
C
      IF(IFLAG(INDEX).EQ.1)GOTO 100
   25 IF(KNT.EQ.0) ITST=INDEX
        IF(ITST.NE.INDEX) THEN
          KNT=0
          GO TO 25
        END IF
   35 IF(WAVE(I).LE.W(INDEX)) THEN
        KNT=KNT+1
        NODE(KNT)=WAVE(I)
        IF(INDEX.EQ.4)NODE(KNT)=NODE(KNT)-1107.
        FUNC(KNT)=SPEC(I)
        GO TO 100
      END IF
C
C     ***************************************************************************
C     *                    COMPUTE LEAST SQUARES FIT.                           *
C     ***************************************************************************
C
   45 IFLAG(INDEX)=1
      EPS=-1.
        IF(INDEX.EQ.4) THEN
          MAXDEG=4
          CALL LEAST(KNT,NODE,FUNC,WEIGHT,EPS,MAXDEG,NDEG,WORK,ANSWER)
          II=INDEX-1
          GOTO 65
        ELSE
          MAXDEG=3
          CALL LEAST(KNT,NODE,FUNC,WEIGHT,EPS,MAXDEG,NDEG,WORK,ANSWER)
          II=INDEX-1
          WAVCAL=QKPRIM(WORK,MAXDEG)
          IF(WAVCAL.GE.W(II).AND.WAVCAL.LE.W(INDEX))GOTO 65
        END IF
C
C     ***************************************************************************
C     *  ROUTINE TO USE INSTEAD OF DERIVATIVE PROCEDURE. THIS SECTION OF  *
C     *  CODE IS NECESSARY SINCE IT IS POSSIBLE FOR THE FIT NOT TO HAVE A *
C     *  DISTINCT MIN OR MAX.                                             *
C     ***************************************************************************
C
```

```
      STEP=(WAVE(2)-WAVE(1))/5.
      WAVNUM=W(II)
      KK=0
   55 IF(WAVNUM.LE.W(INDEX)) THEN
        KK=KK+1
        ABSRB(KK)=EVAL(WAVNUM,MAXDEG,WORK,MAXDEG)
        SIGMA(KK)=WAVNUM
        WAVNUM=WAVNUM+STEP
        GO TO 55
      END IF
      WAVCAL=MAXCAL(SIGMA,ABSRB,KK)
C
C  ************************************************************************
C  *      COMPUTE RESIDUAL STANDARD DEVIATION OF THE FIT AND THE          *
C  *          WAVENUMBERS AT WHICH MAXIMA AND MINIMUM OCCUR.              *
C  ************************************************************************
C
   65 SUM=0.
      DO 70 J=1,KNT
      SUM=SUM+(FUNC(J)-SNGL(ANSWER(J)))**2
   70 CONTINUE
      RESID=SQRT(SUM/FLOAT(KNT-1))
        IF(INDEX.LT.1.OR.INDEX.GT.6) THEN
          GOTO 135
        ELSE IF(INDEX.EQ.2) THEN
          WAVE1=WAVCAL
          MAX1=EVAL(WAVE1,MAXDEG,WORK,MAXDEG)
          R1=RESID
          GO TO 100
        ELSE IF(INDEX.EQ.4) THEN
          CALL PEAKCL(MAXDEG,WORK,WAVEP,MINPK)
          R2=RESID
          GO TO 100
        ELSE IF(INDEX.EQ.6) THEN
          WAVE2=WAVCAL
          MAX2=EVAL(WAVE2,MAXDEG,WORK,MAXDEG)
          R3=RESID
       ELSE
         GO TO 135
       END IF
  100 CONTINUE
      WRITE(30,1015)WAVE1,MAX1,R1,WAVEP,MINPK,R2,WAVE2,MAX2,R3
C
C  ************************************************************************
C  *   CALCULATE THE BASE TRANSMITTANCE AT 1107 WAVENUMBERS, AND THE      *
C  *  ABSORPTION COEFFICIENTS AT THE PEAK AND BASE OF THE ABSORPTION.     *
C  ************************************************************************
C
      BASE=((WAVE2-WAVEP)*MAX1+(WAVEP-WAVE1)*MAX2)/(WAVE2-WAVE1)
      ALFAB=DSPCOR(BASE,TAU)
      ALFAP=DSPCOR(MINPK,TAU)
      ALFNET=ALFAP-ALFAB
C
```

```
C  ***************************************************************************
C  *                    OUTPUT REMAINING RESULTS.                            *
C  ***************************************************************************
C
      NETABS=ALFNET*TAU/2.303
      PPM=(ALFNET-0.5)*6.28
      WRITE(30,1016)WAVEP,BASE
      WRITE(30,1017)ALFAB,ALFAP
      WRITE(30,1018)NETABS
      WRITE(30,1019)PPM
C
C  ***************************************************************************
C  *                      RESET 'IFLAG' ARRAY.                               *
C  ***************************************************************************
C
      DO 110 MM=1,6
  110 IFLAG(MM)=0
C
C  ***************************************************************************
C  *                      CHECK FOR FULL PAGE.                               *
C  ***************************************************************************
C
      NLINES=NLINES+16
        IF((NLINES+16).GE.60) THEN
          NLINES=1
          IPAGE=IPAGE+1
            IF(IPAGE.LT.10) THEN
              WRITE(30,1011)MONTH,DAY,YEAR,IPAGE
            ELSE
              WRITE(30,1012)MONTH,DAY,YEAR,IPAGE
            END IF
        END IF
      GO TO 5
C
C  ***************************************************************************
C  *                  ERROR MESSAGES AND TERMINATION.                        *
C  ***************************************************************************
C
  115 WRITE(6,*) '  ERROR ENCOUNTERED IN READING DATA.'
      GO TO 200
  125 WRITE(6,*) '  WAVENUMBER CALCULATED IS OUTSIDE OF RANGE.'
      WRITE(6,*)INDEX
      NUM=3*MAXDEG
      DO 130 JJ=1,NUM
  130 WRITE(6,*)JJ,WORK(JJ)
      GO TO 200
  135 WRITE(6,*) '  HOW DID YOU CHANGE "INDEX"??????'
  200 STOP
      END
```

```
      FUNCTION DSPCOR(A1,A2)
C
C  ***************************************************************************
C  *       THIS FUNCTION COMPUTES ABSORPTION COEFFICIENT BASED ON             *
C  *                 THE DSP TRANSMITTANCE EQUATION.                          *
C  ***************************************************************************
C
      TERM=(-0.49+SQRT(0.2401+0.36*A1**2))/(0.18*A1)
      DSPCOR=-ALOG10(TERM)*2.303/A2
      RETURN
      END

      REAL FUNCTION MAXCAL(ARRAY1,ARRAY2,NUM)
C
C  ***************************************************************************
C  *     THIS FUNCTION FINDS THE MAXIMUM TRANSMITTANCE WITHIN THE             *
C  *       WAVENUMBER RANGE SPECIFIED BY THE W(II) AND W(INDEX).              *
C  ***************************************************************************
C
      DIMENSION ARRAY1(*),ARRAY2(*)
      PEAK=0.25
      DO 10 KNT=1,NUM
      IF(ARRAY2(KNT).LT.PEAK)GOTO 10
      MAXCAL=ARRAY1(KNT)
      PEAK=ARRAY2(KNT)
   10 CONTINUE
      RETURN
      END

      REAL FUNCTION MINCAL(D)
C
C  ***************************************************************************
C  * WHEN A FOURTH-ORDER POLYNOMIAL IS FIT TO THE OXYGEN TRANSMITTANCE *
C  * MINIMUM, THE RESULTING DERIVATIVE IS A CUBIC EQUATION. THE        *
C  * THE SOLUTION OF THIS EQUATION, IN THE SUBROUTINE 'PEAKCL', IS 3   *
C  * ROOTS. THIS FUNCTION FINDS THE ROOT CLOSEST IN ABSOLUTE VALUE TO  *
C  * 1107 WAVENUMBERS. THIS ROOT SHOULD BE THE WAVENUMBER OF THE       *
C  * INTERSTITIAL OXYGEN MINIMA.                                       *
C  ***************************************************************************
C
      DIMENSION D(3)
      VALUE=ABS(D(1))
      IK=1
      DO 20 KK=2,3
      IF(ABS(D(KK)).LT.VALUE) IK=KK
   20 CONTINUE
      MINCAL=D(IK)
      RETURN
      END
```

```
      SUBROUTINE PEAKCL(NDEG,ARRAY,WAVENM,VALUE)
C
C  ************************************************************************
C  * CODE TO DETERMINE COEFFICIENTS OF THE FOURTH-ORDER POLYNOMIAL      *
C  * FITTED TO THE OXYGEN MINIMUM AND THEN SOLVE THE CUBIC EQUATION     *
C  * FOR THE WAVENUMBER AT THE MINIMUM.                                 *
C  ************************************************************************
C
      REAL MINCAL
      DIMENSION ARRAY(*)
      COEFX3=4.*ARRAY(12)
C
      SUM=0.
        DO 100 JJ=4,1,-1
  100   SUM=SUM+ARRAY(JJ)
      COEFX2=-3.*ARRAY(12)*SUM+3.*ARRAY(11)
C
      SUM1=0.
      SUM2=0.
      SUM3=0.
        DO 200 JJ=4,2,-1
        SUM2=SUM2+ARRAY(JJ+3)
        SUM3=SUM3+ARRAY(JJ-1)
        JJM1=JJ-1
        DO 200 LL=JJM1,1,-1
  200   SUM1=SUM1+ARRAY(JJ)*ARRAY(LL)
      COEFX1=-2.*ARRAY(12)*(SUM1-SUM2)-2.*ARRAY(11)*SUM3+2.*ARRAY(10)
C
      SUM1=0.
      SUM2=0.
      SUM3=0.
      SUM4=0.
        DO 300 JJ=4,3,-1
        SUM2=SUM2+ARRAY(JJ-2)
        SUM3=SUM3+ARRAY(JJ+2)
        JJM1=JJ-1
        DO 300 LL=JJM1,2,-1
        JJM2=JJ-2
        DO 300 MM=JJM2,1,-1
  300   SUM1=ARRAY(JJ)*ARRAY(LL)*ARRAY(MM)
        DO 400 LL=3,2,-1
        LLM1=LL-1
        DO 400 MM=LLM1,1,-1
  400   SUM4=SUM4+ARRAY(LL)*ARRAY(MM)
      COEFX0=-ARRAY(12)*(SUM1-(ARRAY(4)*(ARRAY(5)+ARRAY(6)))+
     *           ARRAY(3)*ARRAY(5)+ARRAY(2)*ARRAY(7)+
     *           (ARRAY(1)*(ARRAY(6)+ARRAY(7))))+ARRAY(11)*(SUM4-SUM3)-
     *           ARRAY(10)*SUM2+ARRAY(9)
C
      A2=COEFX2/COEFX3
      A1=COEFX1/COEFX3
      A0=COEFX0/COEFX3
```

```
      CALL CUBIC(A0,A1,A2,R1,R2,R3)
      ROOT=MINCAL(R1,R2,R3)
      WAVENM=1107.+ROOT
      VALUE=EVAL(ROOT,NDEG,ARRAY,NDEG)
      RETURN
      END

      FUNCTION QKPRIM(ARRAY,NUM)
C
C  ***************************************************************************
C  *      THIS FUNCTION COMPUTES THE WAVENUMBER AT WHICH THE FIRST     *
C  * DERIVATIVE OF THE 3RD-ORDER POLYNOMIAL LEAST SQUARES FIT IS ZERO. *
C  ***************************************************************************
C
      DIMENSION ARRAY(*)
      A=3.*ARRAY(9)
      SUM=0.
      DO 10 II=NUM,1,-1
   10 SUM=SUM+ARRAY(II)
      B=-(2.*SUM*ARRAY(9)-2.*ARRAY(8))
      SUM1=0.
      DO 30 II=NUM,1,-1
      NUM1=II-1
        IF(NUM1.NE.0) THEN
          DO 20 KK=NUM1,1,-1
   20     SUM1=SUM1+ARRAY(II)*ARRAY(KK)
        END IF
   30 CONTINUE
      SUM2=0.
      NUM2=2*NUM-2
      NUM1=NUM+1
      DO 40 II=NUM1,NUM2
   40 SUM2=SUM2+ARRAY(II)
      SUM3=ARRAY(1)+ARRAY(2)
      C=ARRAY(9)*(SUM1-SUM2)-ARRAY(8)*SUM3+ARRAY(7)
      ARGMNT=(B**2)-4.*A*C
        IF(ARGMNT.GE.0) THEN
          TERM1=SQRT(ARGMNT)
          QKPRIM=(-B+TERM1)/(2.*A)
          IF(QKPRIM.LT.0.)QKPRIM=(-B-TERM1)/(2.*A)
          RETURN
        ELSE
          QKPRIM=-1.
          RETURN
        END IF
      END
```

```
      SUBROUTINE LEAST(M,X,F,W,EPS,MAXDEG,NDEG,ARRAY,R)
C
C  THE SUBROUTINE LEAST AND THE FUNCTION EVAL CALCULATE THE LEAST
C  SQUARES POLYNOMIAL APPROXIMATION TO A SET OF DATA SPECIFIED BY THE
C  ARRAY OF M NODES, X, WITH CORRESPONDING FUNCTION VALUES AND WEIGHTS
C  IN THE ARRAYS F AND W, RESPECTIVELY. THE WEIGHTS MUST ALL BE
C  POSITIVE. THE POLYNOMIAL IS DETERMINED IN LEAST AND EVALUATED IN
C  EVAL. ON INPUT EPS IS THE DESIRED WEIGHTED RMS ERROR. THE CODE
C  INCREASES THE DEGREE OF THE FIT IN AN ATTEMPT TO MEET THIS ERROR
C  REQUEST. ON RETURN EPS IS SET TO THE WEIGHTED RMS ERROR OF THE FIT.
C  BECAUSE EPS IS USED FOR BOTH INPUT AND OUTPUT, IT MUST BE A VARIABLE
C  IN THE CALLING PROGRAM. MAXDEG IS THE HIGHEST DEGREE ALLOWED AND MUST
C  BE LESS THAN OR EQUAL TO (M-1). THE ACTUAL DEGREE OF THE FIT IS
C  RETURNED IN NDEG. TO FORCE THE CODE TO USE THE PARTICULAR DEGREE
C  MAXDEG, SET EPS NEGATIVE ON INPUT. THE DOUBLE PRECISION VECTOR R OF M
C  WORDS OUTPUTS THE DOUBLE PRECISION VALUES OF THE POLYNOMIAL FIT AT
C  EACH OF THE DATA POINTS X(I). THE VECTOR "ARRAY" SPECIFIES THE
C  ORTHOGONAL POLYNOMIAL FIT AND PROVIDES WORKING STORAGE. THE DIMENSION
C  OF ARRAY IN THE CALLING PROGRAM MUST BE AT LEAST 2*M+3*MAXDEG. THE
C  ARRAYS X,F,W, ARRAY AND R MUST BE DIMENSION IN THE CALLING PROGRAM.
C
      DIMENSION X(*),F(*),W(*),ARRAY(*)
      DOUBLE PRECISION R(*),SUM,CK,TEMP
C
C  INITIALIZE STORAGE AND CONSTANTS.
C
      IB=MAXDEG+1
      IBL2=MAXDEG-1
      IC=IB+IBL2
      IOL1=IC+MAXDEG
      I1L1=IOL1+M
      RM=M
      TOL=RM*EPS**2
C
C  CALCULATE CONSTANT FIT.
C
      NDEG=0
      S=0.
      SUM=0.0D0
      DO 1 I=1,M
      S=S+W(I)
    1 SUM=SUM+DBLE(W(I))*DBLE(F(I))
      RNO=S
C
C  CK IS THE COEFFICIENT C(0) HERE.
C
      CK=SUM/RNO
      ARRAY(IC)=CK
      ERROR=0.0
      DO 2 I=1,M
      R(I)=CK
    2 ERROR=ERROR+W(I)*SNGL(CK-DBLE(F(I)))**2
      IF(NDEG.EQ.MAXDEG)GO TO 14
      IF(EPS.LT.0.0)GO TO 3
      IF(ERROR.LE.TOL)GO TO 14
C
C  CALCULATE LINEAR FIT.
C
    3 NDEG=1
```

```
      ES=ERROR
      SUM=0.0D0
      DO 4 I=1,M
    4 SUM=SUM+DBLE(W(I))*DBLE(X(I))
C
C  CALCULATE A(1).
C
      ARRAY(1)=SUM/RNO
C
C  CALCULATE Q1(.).
C
      S=0.0
      SUM=0.0D0
      DO 5 I=1,M
      ARRAY(I1L1+I)=X(I)-ARRAY(1)
      S=S+W(I)*ARRAY(I1L1+I)**2
      TEMP=DBLE(F(I))-R(I)
    5 SUM=SUM+DBLE(W(I))*DBLE(ARRAY(I1L1+I))*TEMP
      RN1=S
C
C  CK IS THE COEFFICIENT C(1) HERE.
C
      CK=SUM/RN1
      ARRAY(IC+1)=CK
C
C  CALCULATE THE VALUE OF THE FIT AT THE DATA POINTS AND
C  ALSO THE RMS ERROR.
C
      ERROR=0.0
      DO 6 I=1,M
      R(I)=R(I)+CK*DBLE(ARRAY(I1L1+I))
    6 ERROR=ERROR+W(I)*SNGL(R(I)-DBLE(F(I)))**2
      IF(ERROR.GT.ES.AND.EPS.GE.0.0)GO TO 12
      IF(NDEG.EQ.MAXDEG)GO TO 14
      IF(ERROR.LE.TOL.AND.EPS.GE.0.0)GO TO 14
      DO 7 I=1,M
    7 ARRAY(IOL1+I)=1.0
      NDEG=2
      K=2
C
C  GENERAL FIT.
C
    8 ES=ERROR
C
C  CALCULATE B(K).
C
      ARRAY(IBL2+K)=RN1/RNO
C
C  CALCULATE A(K).
C
      SUM=0.0D0
      DO 9 I=1,M
    9 SUM=SUM+DBLE(W(I))*DBLE(X(I))*DBLE(ARRAY(I1L1+I))**2
      ARRAY(K)=SUM/RN1
C
C  CALCULATE QK(.) OVERWRITING ON QK-2(.).
C
      S=0.0
      SUM=0.0D0
```

```
      DO 10 I=1,M
      ARRAY(IOL1+I)=(X(I)-ARRAY(K))*ARRAY(I1L1+I)
     1  -ARRAY(IBL2+K)*ARRAY(IOL1+I)
      S=S+W(I)*ARRAY(IOL1+I)**2
      TEMP=DBLE(F(I))-R(I)
 10   SUM=SUM+DBLE(W(I))*DBLE(ARRAY(IOL1+I))*TEMP
      RNO=RN1
      RN1=S
C
C  SWAP INDICES SO I1 REFERS TO STORAGE OF QK(.)
C  AND IO TO QK-1(.).
C
      IT=IOL1
      IOL1=I1L1
      I1L1=IT
C
C  CK IS THE COEFFICIENT C(K) HERE.
C
      CK=SUM/RN1
      ARRAY(IC+K)=CK
C
C  CALCULATE THE VALUE OF THE FIT AT THE DATA POINTS AND
C  ALSO THE RMS ERROR.
C
      ERROR=0.0
      DO 11 I=1,M
      R(I)=R(I)+CK*DBLE(ARRAY(I1L1+I))
 11   ERROR=ERROR+W(I)*SNGL(R(I)-DBLE(F(I)))**2
      IF(ERROR.GT.ES.AND.EPS.GE.0.0)GO TO 12
      IF(NDEG.EQ.MAXDEG)GO TO 14
      IF(ERROR.LE.TOL.AND.EPS.GE.0.0)GO TO 14
      NDEG=NDEG+1
      K=K+1
      GO TO 8
C
C  HERE IF ERROR INCREASED ON RAISING DEGREE.
C
 12   NDEG=NDEG-1
      ERROR=ES
      DO 13 I=1,M
 13   R(I)=R(I)-CK*DBLE(ARRAY(I1L1+I))
C
C  EXIT.
C
 14   EPS=SQRT(ERROR/RM)
      RETURN
      END
      FUNCTION EVAL(Y,N,ARRAY,MAXDEG)
C
C  THE FUNCTION EVAL EVALUATES THE ORTHOGONAL POLYNOMIAL
C  FIT COMPUTED BY LEAST AND SPECIFIED BY THE VECTOR
C  ARRAY. THE FIT OF DEGREE N IS EVALUATED AT THE ARGU-
C  MENT Y. N MUST BE LESS THAN OR EQUAL TO NDEG AS
C  RETURNED FROM LEAST. LEAST IS CALLED ONLY ONCE FOR
C  EACH FIT, BUT EVAL IS CALLED ONCE FOR EACH ARGUMENT
C  AT WHICH WE RQUIRE THE VALUE OF THE FIT. MAXDEG MUST
C  HAVE THE SAME VALUE AS IN THE CALL TO LEAST.
C
      DIMENSION ARRAY(1)
```

```
      IB=MAXDEG+1
      IC=MAXDEG+IB-1
C
C  EVALUATE N=0,1 AS SPECIAL CASES.
C
      IF(N.GT.0)GO TO 1
      EVAL=ARRAY(IC)
      RETURN
    1 IF(N.GT.1)GO TO 2
      EVAL=ARRAY(IC)+ARRAY(IC+1)*(Y-ARRAY(1))
      RETURN
C
C  GENERAL RECURRENCE RELATION.
C
    2 DKP2=ARRAY(IC+N)
      DKP1=ARRAY(IC+N-1)+(Y-ARRAY(N))*DKP2
      NL2=N-2
      IF(NL2.LT.1)GO TO 4
      DO 3 L=1,NL2
      K=1+NL2-L
      DK=ARRAY(IC+K)+(Y-ARRAY(K+1))*DKP1
     1   -ARRAY(IB+K)*DKP2
      DKP2=DKP1
    3 DKP1=DK
    4 EVAL=ARRAY(IC)+(Y-ARRAY(1))*DKP1
     1   -ARRAY(IB)*DKP2
      RETURN
      END
```

```
      SUBROUTINE CUBIC(A0,A1,A2,R1,R2,R3)
C
C  ***********************************************************************
C  *            THIS SUBROUTINE WAS WRITTEN BY JERRY LOWNEY              *
C  *                   TO SOLVE THE CUBIC EQUATION:                      *
C  *                  X^3 + A2*X^2 + A1*X + A0   = 0                     *
C  ***********************************************************************
C
      COMPLEX X1,X2,X3,T,U,D,E,F
      P=A2
      Q=A1
      R=A0
      A=1./3*(3*Q-P**2)
      B=1./27*(2*P**3-9*P*Q+27*R)
      D=B**2/4+A**3/27
      E=-B/2+CSQRT(D)
        IF(E.EQ.(0,0)) THEN
          T=0
          GO TO 15
        END IF
        IF(AIMAG(E).EQ.0..AND.REAL(E).LT.0.)THEN
          T=-EXP(ALOG(CABS(E))/3)
          GO TO 15
        END IF
      T=CEXP(CLOG(E)/3)
15    F=-B/2-CSQRT(D)
        IF(F.EQ.(0,0)) THEN
          U=0
          GO TO 16
        END IF
        IF(AIMAG(F).EQ.0..AND.REAL(F).LT.0.)THEN
          U=-EXP(ALOG(CABS(F))/3)
          GO TO 16
        END IF
      U=CEXP(CLOG(F)/3)
16    X1=T+U
      X2=-.5*(T+U)+.5*(T-U)*CSQRT((-3,0))
      X3=-.5*(T+U)-.5*(T-U)*CSQRT((-3,0))
      R1=X1-P/3
      R2=X2-P/3
      R3=X3-P/3
      RETURN
      END
```

Section IIIB. PASCAL Program

```
ASTMDSP  [Pascal Version]

program astmdsp(input,output,fn);

(*F-1 Proposal p 112*)
(*Interstitial Oxygen Content Of Silicon Slices By*)
(*Computer-Assisted Infrated Spectrophotometry*)
(*10/16/85*)
(*Double Side Polished*)
(*-------------------------------------------------------------------------
*) (*------    Modified by Warren Gladden
(*------        date: 8/21/87
(*-------------------------------------------------------------------------
*)

const
  maxr=32;                                    (*max data points in fit*)
maxc=9;                                      (*max number of variables in fit*)

type
  sstring = packed array[1..40] of char;      (*filename type*)
  drayr   = array[1..maxr] of real;           (*array types for fits*)
  drayc   = array[1..maxc] of real;           (*                    *)
  ddray1  = array[1..maxr,1..maxc] of real;   (*                    *)
  ddray2  = array[1..maxc,1..maxc] of real;   (*                    *)

var

  coef,nal                          :drayc;     (*ls fit vars*)
  x,y,yemp,
  dual                              :drayr;     (*ls vars*)
  x1:array[1..1001] of real;                    (*wavenumber array*)
  y1:array[1..1001] of real;                    (*transmittance array*)
  corco,yco,xco                     :real;      (*ls vars*)
  xint,xwave                        :real;      (*interpolate vars*)
  smax2,speak,tmax1,smax1,tmax2     :real;      (*astm vars*)
  tr,thk                            :real;      (*temp trans, thick*)
  rsdmax1,rsdmax2,rsdpeak           :real;      (*residual vars*)
  nmax1,nmax2,npeak                 :real;      (*# pts fit*)
  apeak,abase,tpeak,temp,temp1      :real;      (*abs pk,base*)
  ppm,anet,latnet                   :real;      (*conc var*)
  tbase,tb1,tb2                     :real;      (*baseline var*)
  a1,a2,a3,alphas                   :real;      (*abs coef var*)
  e,del,firstx,roffset              :real;      (*uncertainty var*)
  sumy2,ysum2                       :real;      (*stan. dev. var*)
  nrow,ncol,num,noffset             :integer;   (*array sizes and degree*)
startwave,endwave,wave1,wave2     :integer;   (*wavenumber limit for fit*)
dummy,i,j,xtemp                   :integer;   (*loop and temp. var*)
  fname,fnames,junk                 :string[14];(*filename, ref, samp*)
  fn                                :text;      (*file var*)
(*-------------------------------------------------------------------------
*) procedure base;                       (*baseline @ oxygen*)
begin
  tb1:=smax2-speak;
```

```
  tb2:=speak-smax1;
  tbase:=tb1*tmax1+tb2*tmax2;
  tbase:=tbase/(smax2-smax1);
end;
(*-------------------------------------------------------------------------
*) procedure alpha;                    (*absorption coefficient*)
begin
  a1:=-0.49+sqrt(0.2401+0.36*sqr(tr));
  a2:=a1/(0.18*tr);
  a3:=ln(a2);
  alphas:=-1/thk*a3;
end;
(*-------------------------------------------------------------------------
*) procedure uncertainty;              (*non-symmetrical uncertainty*)
begin
  e:=(rsdmax1/(nmax1-4))+(rsdpeak/(npeak-5))+(rsdmax2/(nmax2-4));
  e:=2*sqrt(e);
end;
(*-------------------------------------------------------------------------
*) procedure standev;                  (*standard deviation*)
begin
  sumy2:=0;
  ysum2:=0;
  j:=0;
  for i:=1 to num do
  begin
    if (x1[i]>=2000) and (x1[i]<=2060) then
    begin
      sumy2:=y1[i]*y1[i]+sumy2;
      ysum2:=y1[i]+ysum2;
      j:=j+1;
    end;
  end;
  ysum2:=sqr(ysum2);
  ysum2:=ysum2/j;
  sumy2:=sumy2-ysum2;
  sumy2:=sumy2/(j-2);
  sumy2:=sqrt(sumy2);
end;
(*-------------------------------------------------------------------------
*) procedure OpenInFile;
var
  OK                                :boolean;
begin
  repeat
    write('Enter name of transmittance file: ');
    readln(fname);
    assign(fn,fname);
    {$I-} reset(fn) {$I-};
    OK:=(IOresult = 0);
    if not OK then
      writeln('Cannot find file: ',fname);
  until OK;
end;
```

```
(*-------------------------------------------------------------------------
*) procedure OpenFile;
begin
  assign(fn,fname);
  rewrite(fn);
end;
(*-------------------------------------------------------------------------
*) procedure sorttrans;              (*move target trans values*)
begin
  for i:=1 to num do
  begin
    if (x1[i]>=wave1) and (x1[i]<=wave2) then
    begin
      if j=1 then startwave:=i;
      x[j]:=x1[i];
      y[j]:=y1[i];
      j:=j+1;
    end;
  end;
  nrow:=j-1;
  endwave:=startwave+j-2;
end;
(*-------------------------------------------------------------------------
*) procedure minmax;                 (*sort trans values*)
begin
  for i:=1 to nrow-1 do
    for j:=i+1 to nrow do
    begin
      if y[i]<y[j] then
      begin
        temp:=y[i]; temp1:=x[i];
        y[i]:=y[j]; x[i]:=x[j];
        y[j]:=temp; x[j]:=temp1;
      end;
    end;
end;
(*-------------------------------------------------------------------------
*) procedure dataout;                (*output fit results to file*)

var
  i                                :integer;

begin
  writeln(fn,wave1,'       to ',wave2);
  writeln(fn);
  writeln(fn,'                y         ycalc    residual');
  writeln(fn);
  for i:=1 to nrow do
    writeln(fn,x[i]:7:4,'   ',y[i]:10:6,'   ',yemp[i]:10:6,dual[i]:9:3);
writeln(fn);
  writeln(fn,'correlation coef is ',corco:8:5);
  writeln(fn);
  for i:=1 to ncol do
    writeln(fn,'coefficient  ',coef[i]);
```

```
  writeln(fn);
end;
(*-------------------------------------------------------------------------
*) procedure s2                          (*least squares fit to # of*)
          (var x:ddray1;                (*x,y points in nrow*)
           var y:drayr;
           var a:ddray2;
           var g:drayc;
           var nrow,ncol:integer);

var
  i,k,l                                 :integer;

begin
  for k:=1 to ncol do
    begin
      for l:=1 to k do
        begin
          a[k,l]:=0;
          for i:=1 to nrow do
            begin
              a[k,l]:=a[k,l]+x[i,l]*x[i,k];
              if k<>l then a[l,k]:=a[k,l]
            end
        end;
      g[k]:=0;
      for i:=1 to nrow do
        g[k]:=g[k]+y[i]*x[i,k]
    end
end;
(*-------------------------------------------------------------------------
*) procedure curve1     (var b       :ddray2;
                        y           :drayc;
                        var coef    :drayc;
                        ncol        :integer);

var
  w                                 :array[1..maxc,1..maxc] of real;
  cfile                             :array[1..maxc,1..maxc] of integer;
  i,j,k,l,nv,rw,cl,n,l1             :integer;
  sure,triad,keep,sum,t,ab,great    :real;
(*-------------------------------------------------------------------------
*) procedure xchng(var  a,b:real);

var
  keep                              :real;

begin
  keep:=a;
  a:=b;
  b:=keep;
end;                                (*swap*)
(*-------------------------------------------------------------------------
*) procedure curve2;
```

```
var
  i,j,k,l,l1                          :integer;
(*-------------------------------------------------------------------------
*) procedure curve3;

var
  l                                   :integer;

begin
  if rw<>cl then
    begin
      sure:=-sure;
      for l:=1 to n do
        xchng(b[rw,l],b[cl,l]);
      if nv>0 then
        for l:=1 to nv do
          xchng(w[rw,l],w[cl,l])
    end
end;

begin
  nv:=1;
  n:=ncol;
  for i:=1 to n do
    begin
      w[i,1]:=y[i];
      cfile[i,3]:=0
    end;
  sure:=1.0;
  for i:=1 to n do
    begin
      great:=0.0;
      for j:=1 to n do
        begin
          if cfile[j,3]<>1 then
            begin
              for k:=1 to n do
                begin
                  if cfile[k,3]<1 then
                    if abs(b[j,k])>great then
                      begin
                        rw:=j;
                        cl:=k;
                        great:=abs(b[j,k])
                      end
                end
            end
    end;
  cfile[cl,3]:=cfile[cl,3]+1;
  cfile[i,1]:=rw;
  cfile[i,2]:=cl;
  curve3;
  triad:=b[cl,cl];
```

```
  sure:=sure*triad;
  b[cl,cl]:=1.0;

  for l:=1 to n do
    b[cl,l]:=b[cl,l]/triad;
  if nv>0 then
    for l:=1 to nv do
      w[cl,l]:=w[cl,l]/triad;

  for l1:=1 to n do
    begin
      if l1<>cl then
        begin
          t:=b[l1,cl];
          b[l1,cl]:=0.0;
          for l:=1 to n do
            b[l1,l]:=b[l1,l]-b[cl,l]*t;
            if nv>0 then
              for l:=1 to nv do
                w[l1,l]:=w[l1,l]-w[cl,l]*t;
        end
    end
  end;
end;

begin
  curve2;
  for i:=1 to n do
    begin
      l:=n-i+1;
      if cfile[l,1]<>cfile[l,2] then
        begin
          rw:=cfile[l,1];
          cl:=cfile[l,2];
          for k:=1 to n do
            xchng(b[k,rw],b[k,cl])
        end
    end;
  for i:=1 to n do
    coef[i]:=w[i,1];
end;
(*-------------------------------------------------------------------------
*) procedure curve(x,y         :drayr;
                  var yemp    :drayr;
                  var dual    :drayr;
                  var coef    :drayc;
                  var nal     :drayc;
                      nrow    :integer;
                  var ncol    :integer);

var
  xmatr                          :ddray1;
  a                              :ddray2;
  g                              :drayc;
```

```
  i,j,nm                          :integer;
  xi,yi,yc,srs,see,sum_y,sum_y2   :real;

begin
  for i:=1 to nrow do
    begin
      xi:=x[i];
      xmatr[i,1]:=1.0;
      for j:=2 to ncol do
        xmatr[i,j]:=xmatr[i,j-1]*xi
      end;
  s2(xmatr,y,a,g,nrow,ncol);
  curve1(a,g,coef,ncol);
  sum_y:=0.0;
  sum_y2:=0.0;
  srs:=0.0;
  for i:=1 to nrow do
    begin
      yi:=y[i];
      yc:=0.0;
      for j:=1 to ncol do
        yc:=yc+coef[j]*xmatr[i,j];
      yemp[i]:=yc;
      dual[i]:=yc-yi;
      srs:=srs+sqr(dual[i]);
      sum_y:=sum_y+yi;
      sum_y2:=sum_y2+yi*yi;
    end;
  corco:=sqrt(1.0-srs/(sum_y2-sqr(sum_y)/nrow));
  if nrow=ncol then nm:=1
  else nm:=nrow-ncol;
  see:=sqrt(srs/nm);
  for i:=1 to ncol do
    nal[i]:=see*sqrt(a[i,i])
  end;
(*-------------------------------------------------------------------------
*) procedure polythree;
begin
  xco:=x[1]-2;
  temp:=0.0;
  temp1:=0.0;
  for i:=1 to 400 do
  begin
    xco:=xco+0.01;
    yco:=coef[1]+coef[2]*xco+coef[3]*sqr(xco)+coef[4]*xco*sqr(xco);
    if yco>temp then
    begin
      temp:=yco;
      temp1:=xco;
    end; (*if*)
  end; (*loop*)
  xtemp:=trunc(temp1);
  xint:=temp1-trunc(temp1);
  xwave:=x1[xtemp+startwave]+xint*1.925;
```

```
end; (*polythree*)
(*-------------------------------------------------------------------------
*) procedure polyfour;
begin
  xco:=x[nrow]-2;
  temp:=10.0;
  temp1:=0.0;
  for i:=1 to 400 do
  begin
    xco:=xco+0.01;
    yco:=coef[1]+coef[2]*xco+coef[3]*sqr(xco)+coef[4]*xco*sqr(xco)+coef[5]*sqr(xco)*
    if yco<temp then
    begin
      temp:=yco;
      temp1:=xco;
    end; (*if*)
  end; (*loop*)
  xtemp:=trunc(temp1);
  xint:=temp1-trunc(temp1);
  xwave:=x1[xtemp+startwave]+xint*1.925;
end; (*polyfour*)
(*-------------------------------------------------------------------------
*) procedure ls1;
begin
  j:=1;
  wave1:=1010;
  wave2:=1060;
  sorttrans;
    for i:=1 to nrow do
    x[i]:=i;
  ncol:=3;
      begin
        ncol:=ncol+1;
        curve(x,y,yemp,dual,coef,nal,nrow,ncol);
        dataout;
      end;
  minmax;
  polythree;
  smax1:=xwave;
  tmax1:=temp;
  rsdmax1:=nal[1];
  nmax1:=nrow;
  writeln(fn);
  writeln(fn,'tmax1       ',tmax1);
  writeln(fn,'smax1       ',smax1);
  writeln(fn,'rsdmax1     ',rsdmax1);
  writeln(fn,'# pts. fit  ',nrow);
  writeln(fn);
end; (*ls1*)
(*-------------------------------------------------------------------------
*) procedure ls2;
begin
  j:=1;
  wave1:=1090;
```

```
  wave2:=1123;
  sorttrans;
    for i:=1 to 17 do
    x[i]:=i;
  ncol:=4;
      begin
        ncol:=ncol+1;
        curve(x,y,yemp,dual,coef,nal,nrow,ncol);
        dataout;
      end;
  minmax;
  polyfour;
  speak:=xwave;
  tpeak:=temp;
  rsdpeak:=nal[1];
  npeak:=nrow;
  writeln(fn);
  writeln(fn,'tpeak       ',tpeak);
  writeln(fn,'speak       ',speak);
  writeln(fn,'rsdpeak     ',rsdpeak);
  writeln(fn,'# pts. fit  ',nrow);
  writeln(fn);
end; (*ls2*)
(*-------------------------------------------------------------------------
*) procedure ls3;
begin
  j:=1;
  wave1:=1200;
  wave2:=1260;
  sorttrans;
    for i:=1 to 32 do
    x[i]:=i;
  ncol:=3;
      begin
        ncol:=ncol+1;
        curve(x,y,yemp,dual,coef,nal,nrow,ncol);
        dataout;
      end;
  minmax;
  polythree;
  smax2:=xwave;
  tmax2:=temp;
  rsdmax2:=nal[1];
  nmax2:=nrow;
  writeln(fn);
  writeln(fn,'tmax2       ',tmax2);
  writeln(fn,'smax2       ',smax2);
  writeln(fn,'rsdmax2     ',rsdmax2);
  writeln(fn,'# pts. fit  ',nrow);
  writeln(fn);
end; (*ls3*)
(*-------------------------------------------------------------------------
*) procedure fileout;                (*output results*)
begin
```

```
  fname:='resltd51.dat';
  OpenFile;
  writeln(fn,'sample file name     ',fnames);
  writeln(fn,'thickness (mils)     ',thk/0.00254);
  writeln(fn);
  writeln(fn,'tbase                ',tbase);
  writeln(fn,'tpeak                ',tpeak);
  writeln(fn,'abase                ',abase);
  writeln(fn,'apeak                ',apeak);
  writeln(fn);
  writeln(fn,'anet                 ',anet);
  writeln(fn,'uncertainty          ',e);
  writeln(fn,'standard deviation   ',sumy2);
  writeln(fn,'ppm  (6.28)          ',ppm);
  close(fn);
end;
(*-------------------------------------------------------------------------
*) begin (*MAIN PROGRAM*)
  OpenInFile;
  fnames:=fname;
    readln(fn,num);
    writeln(num);
    for i:=1 to num do
    begin
      readln(fn,dummy,x1[i],y1[i]);
      if y1[i]>1.0 then y1[i]:=y1[i]/100;
    end;
  close(fn);
  del:=x1[2]-x1[1];
  firstx:=x1[1];
  roffset:=(firstx/del);                    (* noffset:=int(roffset);*)
writeln(roffset,del,firstx);
  writeln('thickness (cm) ');
  readln(thk);
  thk:=thk*1.0;

    fname:='least.dat';
    OpenFile;
    writeln(fn,'sample file name    ',fnames);
    writeln(fn,'thickness (cm)    ',thk/1.0);
    writeln(fn);
    ls1;
    ls2;
    ls3;
  close(fn);
  base;
  tr:=tbase;
  alpha;
  abase:=alphas;
  tr:=tpeak;
  alpha;
  apeak:=alphas;
  anet:=apeak-abase;
  latnet:=anet-0.5;
```

```
  ppm:=latnet*6.28;
  uncertainty;
  standev;
    writeln('OXYGEN PPM  (6.28)  ',ppm);
    writeln;
    writeln('UNCERTAINTY         ',e);
    writeln;
    writeln('STANDARD DEVIATION  ',sumy2);
  fileout;
end.
```

Section IV. SAMPLE DATA

The data are listed in three columns, with two sets of three columns per page. The first column is just an index number. The second column lists the wavenumber, and the third column lists the transmittance at that wavenumber. The single integer number at the head of each spectrum is the number of data points in that spectrum.

Sample: D07

840

261	501.401	0.46960
262	503.330	0.46700
263	505.258	0.46442
264	507.187	0.46066
265	509.115	0.45473
266	511.044	0.44725
267	512.972	0.44130
268	514.901	0.44220
269	516.829	0.44576
270	518.758	0.44822
271	520.686	0.45390
272	522.615	0.45969
273	524.543	0.46176
274	526.471	0.46235
275	528.400	0.46203
276	530.328	0.46114
277	532.257	0.46034
278	534.185	0.45943
279	536.114	0.45866
280	538.042	0.45804
281	539.971	0.45698
282	541.899	0.45474
283	543.828	0.45230
284	545.756	0.45063
285	547.685	0.44929
286	549.613	0.44864
287	551.542	0.44810
288	553.470	0.44630
289	555.398	0.44405
290	557.327	0.44202
291	559.255	0.44026
292	561.184	0.43863
293	563.112	0.43723
294	565.041	0.43681
295	566.969	0.43722
296	568.898	0.43896
297	570.826	0.44246
298	572.755	0.44637
299	574.683	0.44892
300	576.612	0.44954
301	578.540	0.44934
302	580.469	0.44890
303	582.397	0.44707
304	584.326	0.44355
305	586.254	0.43886
306	588.182	0.43340
307	590.111	0.42661
308	592.039	0.41782
309	593.968	0.40774
310	595.896	0.39597
311	597.825	0.38115
312	599.753	0.36256
313	601.682	0.34260
314	603.610	0.32663
315	605.539	0.31448
316	607.467	0.30306
317	609.396	0.29526
318	611.324	0.29407
319	613.253	0.29794
320	615.181	0.30223
321	617.109	0.30424
322	619.038	0.30586
323	620.966	0.31074
324	622.895	0.32582
325	624.823	0.35559
326	626.752	0.39533
327	628.680	0.43551
328	630.609	0.46502
329	632.537	0.47996
330	634.466	0.48596
331	636.394	0.48837
332	638.323	0.49016
333	640.251	0.49234
334	642.180	0.49368
335	644.108	0.49471
336	646.036	0.49555
337	647.965	0.49591
338	649.893	0.49675
339	651.822	0.49760
340	653.750	0.49810
341	655.679	0.49854
342	657.607	0.49898
343	659.536	0.49959
344	661.464	0.49968
345	663.393	0.49990
346	665.321	0.49975
347	667.250	0.49796
348	669.178	0.49785
349	671.107	0.50024
350	673.035	0.50163
351	674.963	0.50216
352	676.892	0.50297
353	678.820	0.50352
354	680.749	0.50305
355	682.677	0.50219
356	684.606	0.50201
357	686.534	0.50235
358	688.463	0.50276
359	690.391	0.50286
360	692.320	0.50254
361	694.248	0.50323
362	696.177	0.50546
363	698.105	0.50767
364	700.034	0.50817
365	701.962	0.50730
366	703.890	0.50631
367	705.819	0.50468
368	707.747	0.50161
369	709.676	0.49768
370	711.604	0.49243
371	713.533	0.48546
372	715.461	0.47828
373	717.390	0.47242
374	719.318	0.46809
375	721.247	0.46433
376	723.175	0.46016

377	725.104	0.45587
378	727.032	0.45224
379	728.961	0.44886
380	730.889	0.44554
381	732.817	0.44276
382	734.746	0.43980
383	736.674	0.43686
384	738.603	0.43605
385	740.531	0.43816
386	742.460	0.44186
387	744.388	0.44549
388	746.317	0.44848
389	748.245	0.45117
390	750.174	0.45307
391	752.102	0.45431
392	754.031	0.45566
393	755.959	0.45651
394	757.888	0.45704
395	759.816	0.45794
396	761.745	0.45926
397	763.673	0.46062
398	765.601	0.46116
399	767.530	0.46109
400	769.458	0.46123
401	771.387	0.46151
402	773.315	0.46171
403	775.244	0.46163
404	777.172	0.46167
405	779.101	0.46222
406	781.029	0.46290
407	782.958	0.46355
408	784.886	0.46413
409	786.815	0.46439
410	788.743	0.46463
411	790.672	0.46507
412	792.600	0.46578
413	794.528	0.46716
414	796.457	0.46973
415	798.385	0.47269
416	800.314	0.47414
417	802.242	0.47344
418	804.171	0.47130
419	806.099	0.46875
420	808.028	0.46692
421	809.956	0.46560
422	811.885	0.46432
423	813.813	0.46300
424	815.742	0.46196
425	817.670	0.46176
426	819.599	0.46219
427	821.527	0.46323
428	823.455	0.46511
429	825.384	0.46756
430	827.312	0.47035
431	829.241	0.47370
432	831.169	0.47722
433	833.098	0.47986
434	835.026	0.48132
435	836.955	0.48239
436	838.883	0.48299
437	840.812	0.48252
438	842.740	0.48117
439	844.669	0.47958
440	846.597	0.47846
441	848.526	0.47769
442	850.454	0.47675
443	852.382	0.47558
444	854.311	0.47381
445	856.239	0.47217
446	858.168	0.47083
447	860.096	0.46848
448	862.025	0.46577
449	863.953	0.46384
450	865.882	0.46251
451	867.810	0.46162
452	869.739	0.46074
453	871.667	0.45976
454	873.596	0.45898
455	875.524	0.45840
456	877.453	0.45814
457	879.381	0.45767
458	881.309	0.45655
459	883.238	0.45538
460	885.166	0.45449
461	887.095	0.45431
462	889.023	0.45481
463	890.952	0.45488
464	892.880	0.45447
465	894.809	0.45435
466	896.737	0.45475
467	898.666	0.45508
468	900.594	0.45530
469	902.523	0.45612
470	904.451	0.45761
471	906.380	0.45901
472	908.308	0.46016
473	910.236	0.46137
474	912.165	0.46300
475	914.093	0.46478
476	916.022	0.46641
477	917.950	0.46779
478	919.879	0.46885
479	921.807	0.46997
480	923.736	0.47143
481	925.664	0.47271
482	927.593	0.47343
483	929.521	0.47395
484	931.450	0.47474
485	933.378	0.47570
486	935.307	0.47667
487	937.235	0.47742
488	939.163	0.47766
489	941.092	0.47777
490	943.020	0.47796
491	944.949	0.47769
492	946.877	0.47711
493	948.806	0.47686
494	950.734	0.47682

495	952.663	0.47634
496	954.591	0.47555
497	956.520	0.47529
498	958.448	0.47553
499	960.377	0.47556
500	962.305	0.47536
501	964.234	0.47537
502	966.162	0.47577
503	968.090	0.47627
504	970.019	0.47666
505	971.947	0.47733
506	973.876	0.47826
507	975.804	0.47913
508	977.733	0.47996
509	979.661	0.48104
510	981.590	0.48221
511	983.518	0.48332
512	985.447	0.48455
513	987.375	0.48575
514	989.304	0.48692
515	991.232	0.48838
516	993.161	0.48969
517	995.089	0.49079
518	997.017	0.49185
519	998.946	0.49266
520	1000.874	0.49337
521	1002.803	0.49408
522	1004.731	0.49472
523	1006.660	0.49550
524	1008.588	0.49630
525	1010.517	0.49681
526	1012.445	0.49720
527	1014.374	0.49795
528	1016.302	0.49895
529	1018.231	0.49967
530	1020.159	0.50038
531	1022.088	0.50105
532	1024.016	0.50145
533	1025.944	0.50199
534	1027.873	0.50242
535	1029.801	0.50244
536	1031.730	0.50266
537	1033.658	0.50295
538	1035.587	0.50319
539	1037.515	0.50343
540	1039.444	0.50351
541	1041.372	0.50333
542	1043.301	0.50303
543	1045.229	0.50285
544	1047.158	0.50273
545	1049.086	0.50239
546	1051.015	0.50183
547	1052.943	0.50132
548	1054.871	0.50111
549	1056.800	0.50069
550	1058.728	0.49976
551	1060.657	0.49881
552	1062.585	0.49780
553	1064.514	0.49681
554	1066.442	0.49577
555	1068.371	0.49413
556	1070.299	0.49227
557	1072.228	0.49030
558	1074.156	0.48766
559	1076.085	0.48458
560	1078.013	0.48105
561	1079.942	0.47694
562	1081.870	0.47217
563	1083.798	0.46656
564	1085.727	0.46026
565	1087.655	0.45329
566	1089.584	0.44575
567	1091.512	0.43788
568	1093.441	0.42984
569	1095.369	0.42187
570	1097.298	0.41427
571	1099.226	0.40762
572	1101.155	0.40221
573	1103.083	0.39798
574	1105.012	0.39523
575	1106.940	0.39405
576	1108.869	0.39428
577	1110.797	0.39610
578	1112.725	0.39932
579	1114.654	0.40384
580	1116.582	0.40977
581	1118.511	0.41668
582	1120.439	0.42433
583	1122.368	0.43274
584	1124.296	0.44144
585	1126.225	0.45002
586	1128.153	0.45796
587	1130.082	0.46495
588	1132.010	0.47091
589	1133.939	0.47589
590	1135.867	0.47999
591	1137.796	0.48322
592	1139.724	0.48583
593	1141.652	0.48783
594	1143.581	0.48921
595	1145.509	0.49030
596	1147.438	0.49132
597	1149.366	0.49233
598	1151.295	0.49310
599	1153.223	0.49364
600	1155.152	0.49431
601	1157.080	0.49511
602	1159.009	0.49594
603	1160.937	0.49662
604	1162.866	0.49698
605	1164.794	0.49730
606	1166.723	0.49781
607	1168.651	0.49837
608	1170.579	0.49881
609	1172.508	0.49930
610	1174.436	0.49998
611	1176.365	0.50071
612	1178.293	0.50142

613	1180.222	0.50189
614	1182.150	0.50200
615	1184.079	0.50221
616	1186.007	0.50274
617	1187.936	0.50346
618	1189.864	0.50419
619	1191.793	0.50478
620	1193.721	0.50539
621	1195.650	0.50601
622	1197.578	0.50655
623	1199.506	0.50716
624	1201.435	0.50776
625	1203.363	0.50819
626	1205.292	0.50861
627	1207.220	0.50901
628	1209.149	0.50933
629	1211.077	0.50969
630	1213.006	0.50997
631	1214.934	0.51005
632	1216.863	0.50998
633	1218.791	0.51005
634	1220.720	0.51017
635	1222.648	0.51010
636	1224.577	0.50998
637	1226.505	0.50992
638	1228.433	0.50992
639	1230.362	0.50994
640	1232.290	0.51012
641	1234.219	0.51048
642	1236.147	0.51074
643	1238.076	0.51078
644	1240.004	0.51074
645	1241.933	0.51070
646	1243.861	0.51064
647	1245.790	0.51059
648	1247.718	0.51066
649	1249.647	0.51079
650	1251.575	0.51089
651	1253.504	0.51082
652	1255.432	0.51071
653	1257.360	0.51073
654	1259.289	0.51092
655	1261.217	0.51102
656	1263.146	0.51093
657	1265.074	0.51068
658	1267.003	0.51056
659	1268.931	0.51055
660	1270.860	0.51057
661	1272.788	0.51048
662	1274.717	0.51019
663	1276.645	0.50994
664	1278.574	0.50988
665	1280.502	0.50955
666	1282.431	0.50910
667	1284.359	0.50887
668	1286.287	0.50878
669	1288.216	0.50850
670	1290.144	0.50814
671	1292.073	0.50792
672	1294.001	0.50787
673	1295.930	0.50792
674	1297.858	0.50812
675	1299.787	0.50813
676	1301.715	0.50794
677	1303.644	0.50810
678	1305.572	0.50853
679	1307.501	0.50889
680	1309.429	0.50915
681	1311.358	0.50936
682	1313.286	0.50971
683	1315.214	0.51023
684	1317.143	0.51061
685	1319.071	0.51082
686	1321.000	0.51122
687	1322.928	0.51174
688	1324.857	0.51212
689	1326.785	0.51265
690	1328.714	0.51327
691	1330.642	0.51348
692	1332.571	0.51350
693	1334.499	0.51353
694	1336.428	0.51361
695	1338.356	0.51396
696	1340.285	0.51428
697	1342.213	0.51443
698	1344.141	0.51459
699	1346.070	0.51465
700	1347.998	0.51461
701	1349.927	0.51454
702	1351.855	0.51443
703	1353.784	0.51436
704	1355.712	0.51432
705	1357.641	0.51420
706	1359.569	0.51402
707	1361.498	0.51388
708	1363.426	0.51366
709	1365.355	0.51346
710	1367.283	0.51335
711	1369.212	0.51327
712	1371.140	0.51330
713	1373.068	0.51328
714	1374.997	0.51315
715	1376.925	0.51315
716	1378.854	0.51337
717	1380.782	0.51381
718	1382.711	0.51427
719	1384.639	0.51427
720	1386.568	0.51387
721	1388.496	0.51363
722	1390.425	0.51374
723	1392.353	0.51375
724	1394.282	0.51365
725	1396.210	0.51381
726	1398.139	0.51402
727	1400.067	0.51414
728	1401.995	0.51427
729	1403.924	0.51414
730	1405.852	0.51394

731	1407.781	0.51388
732	1409.709	0.51374
733	1411.638	0.51358
734	1413.566	0.51347
735	1415.495	0.51326
736	1417.423	0.51295
737	1419.352	0.51267
738	1421.280	0.51257
739	1423.209	0.51238
740	1425.137	0.51191
741	1427.066	0.51141
742	1428.994	0.51095
743	1430.922	0.51048
744	1432.851	0.51000
745	1434.779	0.50952
746	1436.708	0.50900
747	1438.636	0.50851
748	1440.565	0.50808
749	1442.493	0.50774
750	1444.422	0.50751
751	1446.350	0.50751
752	1448.279	0.50763
753	1450.207	0.50780
754	1452.136	0.50819
755	1454.064	0.50864
756	1455.993	0.50903
757	1457.921	0.50944
758	1459.849	0.51000
759	1461.778	0.51082
760	1463.706	0.51173
761	1465.635	0.51277
762	1467.563	0.51373
763	1469.492	0.51440
764	1471.420	0.51513
765	1473.349	0.51603
766	1475.277	0.51696
767	1477.206	0.51785
768	1479.134	0.51858
769	1481.063	0.51921
770	1482.991	0.51980
771	1484.920	0.52036
772	1486.848	0.52092
773	1488.777	0.52136
774	1490.705	0.52176
775	1492.634	0.52206
776	1494.562	0.52220
777	1496.490	0.52234
778	1498.419	0.52277
779	1500.347	0.52334
780	1502.276	0.52356
781	1504.204	0.52333
782	1506.133	0.52331
783	1508.061	0.52369
784	1509.990	0.52403
785	1511.918	0.52416
786	1513.847	0.52423
787	1515.775	0.52438
788	1517.704	0.52441
789	1519.632	0.52430
790	1521.561	0.52416
791	1523.489	0.52417
792	1525.417	0.52428
793	1527.346	0.52438
794	1529.274	0.52457
795	1531.203	0.52483
796	1533.131	0.52492
797	1535.060	0.52489
798	1536.988	0.52458
799	1538.917	0.52418
800	1540.845	0.52420
801	1542.774	0.52441
802	1544.702	0.52452
803	1546.631	0.52467
804	1548.559	0.52470
805	1550.488	0.52466
806	1552.416	0.52489
807	1554.344	0.52515
808	1556.273	0.52492
809	1558.201	0.52432
810	1560.130	0.52407
811	1562.058	0.52440
812	1563.987	0.52477
813	1565.915	0.52494
814	1567.844	0.52497
815	1569.772	0.52493
816	1571.701	0.52489
817	1573.629	0.52490
818	1575.558	0.52494
819	1577.486	0.52512
820	1579.415	0.52523
821	1581.343	0.52513
822	1583.271	0.52508
823	1585.200	0.52516
824	1587.128	0.52524
825	1589.057	0.52529
826	1590.985	0.52518
827	1592.914	0.52508
828	1594.842	0.52518
829	1596.771	0.52524
830	1598.699	0.52525
831	1600.628	0.52528
832	1602.556	0.52530
833	1604.485	0.52532
834	1606.413	0.52535
835	1608.342	0.52531
836	1610.270	0.52530
837	1612.198	0.52528
838	1614.127	0.52519
839	1616.055	0.52514
840	1617.984	0.52536
841	1619.912	0.52554
842	1621.841	0.52531
843	1623.769	0.52509
844	1625.698	0.52523
845	1627.626	0.52535
846	1629.555	0.52542
847	1631.483	0.52549
848	1633.412	0.52546

849	1635.340	0.52541
850	1637.269	0.52558
851	1639.197	0.52578
852	1641.125	0.52573
853	1643.054	0.52561
854	1644.982	0.52551
855	1646.911	0.52546
856	1648.839	0.52557
857	1650.768	0.52550
858	1652.696	0.52520
859	1654.625	0.52530
860	1656.553	0.52567
861	1658.482	0.52583
862	1660.410	0.52572
863	1662.339	0.52553
864	1664.267	0.52555
865	1666.196	0.52571
866	1668.124	0.52583
867	1670.052	0.52593
868	1671.981	0.52602
869	1673.909	0.52604
870	1675.838	0.52602
871	1677.766	0.52597
872	1679.695	0.52602
873	1681.623	0.52600
874	1683.552	0.52571
875	1685.480	0.52562
876	1687.409	0.52592
877	1689.337	0.52612
878	1691.266	0.52609
879	1693.194	0.52597
880	1695.123	0.52575
881	1697.051	0.52559
882	1698.979	0.52543
883	1700.908	0.52525
884	1702.836	0.52522
885	1704.765	0.52516
886	1706.693	0.52483
887	1708.622	0.52461
888	1710.550	0.52452
889	1712.479	0.52444
890	1714.407	0.52427
891	1716.336	0.52381
892	1718.264	0.52349
893	1720.193	0.52360
894	1722.121	0.52376
895	1724.050	0.52372
896	1725.978	0.52366
897	1727.906	0.52365
898	1729.835	0.52389
899	1731.763	0.52414
900	1733.692	0.52427
901	1735.620	0.52457
902	1737.549	0.52496
903	1739.477	0.52516
904	1741.406	0.52540
905	1743.334	0.52581
906	1745.263	0.52608
907	1747.191	0.52624
908	1749.120	0.52641
909	1751.048	0.52664
910	1752.977	0.52691
911	1754.905	0.52717
912	1756.833	0.52733
913	1758.762	0.52746
914	1760.690	0.52749
915	1762.619	0.52755
916	1764.547	0.52766
917	1766.476	0.52769
918	1768.404	0.52766
919	1770.333	0.52753
920	1772.261	0.52748
921	1774.190	0.52757
922	1776.118	0.52767
923	1778.047	0.52772
924	1779.975	0.52768
925	1781.904	0.52766
926	1783.832	0.52773
927	1785.760	0.52784
928	1787.689	0.52794
929	1789.617	0.52785
930	1791.546	0.52773
931	1793.474	0.52771
932	1795.403	0.52773
933	1797.331	0.52777
934	1799.260	0.52783
935	1801.188	0.52790
936	1803.117	0.52797
937	1805.045	0.52794
938	1806.974	0.52784
939	1808.902	0.52778
940	1810.831	0.52785
941	1812.759	0.52798
942	1814.688	0.52802
943	1816.616	0.52809
944	1818.544	0.52824
945	1820.473	0.52826
946	1822.401	0.52815
947	1824.330	0.52799
948	1826.258	0.52797
949	1828.187	0.52810
950	1830.115	0.52821
951	1832.044	0.52829
952	1833.972	0.52825
953	1835.901	0.52814
954	1837.829	0.52820
955	1839.758	0.52837
956	1841.686	0.52840
957	1843.615	0.52847
958	1845.543	0.52856
959	1847.471	0.52862
960	1849.400	0.52863
961	1851.328	0.52851
962	1853.257	0.52835
963	1855.185	0.52831
964	1857.114	0.52848
965	1859.042	0.52873
966	1860.971	0.52892

967	1862.899	0.52900
968	1864.828	0.52891
969	1866.756	0.52876
970	1868.685	0.52880
971	1870.613	0.52895
972	1872.542	0.52912
973	1874.470	0.52917
974	1876.398	0.52906
975	1878.327	0.52900
976	1880.255	0.52904
977	1882.184	0.52903
978	1884.112	0.52912
979	1886.041	0.52927
980	1887.969	0.52925
981	1889.898	0.52913
982	1891.826	0.52917
983	1893.755	0.52929
984	1895.683	0.52937
985	1897.612	0.52940
986	1899.540	0.52932
987	1901.469	0.52927
988	1903.397	0.52930
989	1905.325	0.52938
990	1907.254	0.52949
991	1909.182	0.52949
992	1911.111	0.52945
993	1913.039	0.52949
994	1914.968	0.52964
995	1916.896	0.52978
996	1918.825	0.52970
997	1920.753	0.52958
998	1922.682	0.52966
999	1924.610	0.52982
1000	1926.539	0.52987
1001	1928.467	0.52982
1002	1930.396	0.52986
1003	1932.324	0.52997
1004	1934.252	0.53007
1005	1936.181	0.53021
1006	1938.109	0.53027
1007	1940.038	0.53016
1008	1941.966	0.53013
1009	1943.895	0.53010
1010	1945.823	0.52996
1011	1947.752	0.52991
1012	1949.680	0.53010
1013	1951.609	0.53036
1014	1953.537	0.53047
1015	1955.466	0.53043
1016	1957.394	0.53035
1017	1959.323	0.53033
1018	1961.251	0.53038
1019	1963.179	0.53050
1020	1965.108	0.53067
1021	1967.036	0.53073
1022	1968.965	0.53059
1023	1970.893	0.53056
1024	1972.822	0.53057
1025	1974.750	0.53064
1026	1976.679	0.53081
1027	1978.607	0.53083
1028	1980.536	0.53074
1029	1982.464	0.53066
1030	1984.393	0.53066
1031	1986.321	0.53079
1032	1988.250	0.53090
1033	1990.178	0.53101
1034	1992.106	0.53111
1035	1994.035	0.53121
1036	1995.963	0.53122
1037	1997.892	0.53115
1038	1999.820	0.53113
1039	2001.749	0.53108
1040	2003.677	0.53094
1041	2005.606	0.53087
1042	2007.534	0.53097
1043	2009.463	0.53124
1044	2011.391	0.53145
1045	2013.320	0.53149
1046	2015.248	0.53156
1047	2017.177	0.53158
1048	2019.105	0.53143
1049	2021.033	0.53140
1050	2022.962	0.53151
1051	2024.890	0.53147
1052	2026.819	0.53134
1053	2028.747	0.53135
1054	2030.676	0.53166
1055	2032.604	0.53183
1056	2034.533	0.53177
1057	2036.461	0.53170
1058	2038.390	0.53165
1059	2040.318	0.53159
1060	2042.247	0.53147
1061	2044.175	0.53145
1062	2046.104	0.53161
1063	2048.032	0.53181
1064	2049.960	0.53186
1065	2051.889	0.53190
1066	2053.817	0.53204
1067	2055.746	0.53214
1068	2057.674	0.53204
1069	2059.603	0.53186
1070	2061.531	0.53179
1071	2063.460	0.53177
1072	2065.388	0.53176
1073	2067.317	0.53199
1074	2069.245	0.53225
1075	2071.174	0.53222
1076	2073.102	0.53220
1077	2075.031	0.53215
1078	2076.959	0.53206
1079	2078.887	0.53208
1080	2080.816	0.53220
1081	2082.744	0.53221
1082	2084.673	0.53223
1083	2086.601	0.53242
1084	2088.530	0.53262

1085	2090.458	0.53247
1086	2092.387	0.53230
1087	2094.315	0.53235
1088	2096.244	0.53232
1089	2098.172	0.53228
1090	2100.101	0.53234
1091	2102.029	0.53239
1092	2103.958	0.53249
1093	2105.886	0.53258
1094	2107.814	0.53258
1095	2109.743	0.53251
1096	2111.671	0.53237
1097	2113.600	0.53240
1098	2115.528	0.53230
1099	2117.457	0.53218
1100	2119.385	0.53245

Sample: D12

602		
501	964.234	0.49003
502	966.162	0.49035
503	968.090	0.49108
504	970.019	0.49175
505	971.947	0.49243
506	973.876	0.49335
507	975.804	0.49438
508	977.733	0.49546
509	979.661	0.49669
510	981.590	0.49791
511	983.518	0.49917
512	985.447	0.50055
513	987.375	0.50208
514	989.304	0.50354
515	991.232	0.50485
516	993.161	0.50603
517	995.089	0.50711
518	997.017	0.50840
519	998.946	0.50978
520	1000.874	0.51085
521	1002.803	0.51183
522	1004.731	0.51270
523	1006.660	0.51347
524	1008.588	0.51419
525	1010.517	0.51474
526	1012.445	0.51517
527	1014.374	0.51566
528	1016.302	0.51649
529	1018.231	0.51759
530	1020.159	0.51850
531	1022.088	0.51906
532	1024.016	0.51951
533	1025.944	0.52010
534	1027.873	0.52069
535	1029.801	0.52098
536	1031.730	0.52127
537	1033.658	0.52176
538	1035.587	0.52204
539	1037.515	0.52196
540	1039.444	0.52182
541	1041.372	0.52187
542	1043.301	0.52206
543	1045.229	0.52203
544	1047.158	0.52181
545	1049.086	0.52134
546	1051.015	0.52066
547	1052.943	0.52016
548	1054.871	0.51983
549	1056.800	0.51926
550	1058.728	0.51851
551	1060.657	0.51772
552	1062.585	0.51680
553	1064.514	0.51554
554	1066.442	0.51402
555	1068.371	0.51246
556	1070.299	0.51061
557	1072.228	0.50825
558	1074.156	0.50541
559	1076.085	0.50212
560	1078.013	0.49817
561	1079.942	0.49355
562	1081.870	0.48833
563	1083.798	0.48228
564	1085.727	0.47535
565	1087.655	0.46771
566	1089.584	0.45924
567	1091.512	0.45028
568	1093.441	0.44132
569	1095.369	0.43255
570	1097.298	0.42423
571	1099.226	0.41666
572	1101.155	0.41035
573	1103.083	0.40570
574	1105.012	0.40268
575	1106.940	0.40144
576	1108.869	0.40198
577	1110.797	0.40421
578	1112.725	0.40799
579	1114.654	0.41300
580	1116.582	0.41922
581	1118.511	0.42687
582	1120.439	0.43576
583	1122.368	0.44555
584	1124.296	0.45540
585	1126.225	0.46488
586	1128.153	0.47408
587	1130.082	0.48238
588	1132.010	0.48932
589	1133.939	0.49504
590	1135.867	0.49954
591	1137.796	0.50286
592	1139.724	0.50550
593	1141.652	0.50780
594	1143.581	0.50968
595	1145.509	0.51125
596	1147.438	0.51266
597	1149.366	0.51379
598	1151.295	0.51458
599	1153.223	0.51527
600	1155.152	0.51620
601	1157.080	0.51706
602	1159.009	0.51770
603	1160.937	0.51819
604	1162.866	0.51881
605	1164.794	0.51940
606	1166.723	0.51990
607	1168.651	0.52043
608	1170.579	0.52111
609	1172.508	0.52201
610	1174.436	0.52286
611	1176.365	0.52341
612	1178.293	0.52383
613	1180.222	0.52428
614	1182.150	0.52474
615	1184.079	0.52537
616	1186.007	0.52606

617	1187.936	0.52678
618	1189.864	0.52753
619	1191.793	0.52825
620	1193.721	0.52882
621	1195.650	0.52936
622	1197.578	0.53005
623	1199.506	0.53079
624	1201.435	0.53141
625	1203.363	0.53195
626	1205.292	0.53239
627	1207.220	0.53268
628	1209.149	0.53300
629	1211.077	0.53342
630	1213.006	0.53383
631	1214.934	0.53405
632	1216.863	0.53399
633	1218.791	0.53379
634	1220.720	0.53368
635	1222.648	0.53374
636	1224.577	0.53388
637	1226.505	0.53402
638	1228.433	0.53406
639	1230.362	0.53396
640	1232.290	0.53403
641	1234.219	0.53435
642	1236.147	0.53462
643	1238.076	0.53467
644	1240.004	0.53460
645	1241.933	0.53453
646	1243.861	0.53447
647	1245.790	0.53442
648	1247.718	0.53456
649	1249.647	0.53474
650	1251.575	0.53479
651	1253.504	0.53465
652	1255.432	0.53449
653	1257.360	0.53441
654	1259.289	0.53439
655	1261.217	0.53436
656	1263.146	0.53433
657	1265.074	0.53433
658	1267.003	0.53433
659	1268.931	0.53423
660	1270.860	0.53397
661	1272.788	0.53355
662	1274.717	0.53312
663	1276.645	0.53289
664	1278.574	0.53284
665	1280.502	0.53269
666	1282.431	0.53239
667	1284.359	0.53196
668	1286.287	0.53159
669	1288.216	0.53128
670	1290.144	0.53106
671	1292.073	0.53107
672	1294.001	0.53114
673	1295.930	0.53099
674	1297.858	0.53085
675	1299.787	0.53077
676	1301.715	0.53080
677	1303.644	0.53115
678	1305.572	0.53159
679	1307.501	0.53176
680	1309.429	0.53189
681	1311.358	0.53214
682	1313.286	0.53247
683	1315.214	0.53286
684	1317.143	0.53335
685	1319.071	0.53383
686	1321.000	0.53430
687	1322.928	0.53471
688	1324.857	0.53496
689	1326.785	0.53531
690	1328.714	0.53590
691	1330.642	0.53633
692	1332.571	0.53654
693	1334.499	0.53661
694	1336.428	0.53672
695	1338.356	0.53707
696	1340.285	0.53745
697	1342.213	0.53748
698	1344.141	0.53735
699	1346.070	0.53727
700	1347.998	0.53731
701	1349.927	0.53732
702	1351.855	0.53719
703	1353.784	0.53704
704	1355.712	0.53698
705	1357.641	0.53679
706	1359.569	0.53635
707	1361.498	0.53606
708	1363.426	0.53602
709	1365.355	0.53595
710	1367.283	0.53579
711	1369.212	0.53559
712	1371.140	0.53538
713	1373.068	0.53531
714	1374.997	0.53547
715	1376.925	0.53555
716	1378.854	0.53563
717	1380.782	0.53564
718	1382.711	0.53546
719	1384.639	0.53521
720	1386.568	0.53517
721	1388.496	0.53539
722	1390.425	0.53557
723	1392.353	0.53563
724	1394.282	0.53583
725	1396.210	0.53605
726	1398.139	0.53612
727	1400.067	0.53617
728	1401.995	0.53630
729	1403.924	0.53639
730	1405.852	0.53632
731	1407.781	0.53615
732	1409.709	0.53609
733	1411.638	0.53599
734	1413.566	0.53572

735	1415.495	0.53528
736	1417.423	0.53492
737	1419.352	0.53474
738	1421.280	0.53462
739	1423.209	0.53433
740	1425.137	0.53399
741	1427.066	0.53362
742	1428.994	0.53308
743	1430.922	0.53246
744	1432.851	0.53185
745	1434.779	0.53114
746	1436.708	0.53061
747	1438.636	0.53034
748	1440.565	0.53010
749	1442.493	0.52983
750	1444.422	0.52963
751	1446.350	0.52947
752	1448.279	0.52936
753	1450.207	0.52941
754	1452.136	0.52988
755	1454.064	0.53044
756	1455.993	0.53074
757	1457.921	0.53128
758	1459.849	0.53221
759	1461.778	0.53308
760	1463.706	0.53404
761	1465.635	0.53503
762	1467.563	0.53600
763	1469.492	0.53699
764	1471.420	0.53773
765	1473.349	0.53840
766	1475.277	0.53937
767	1477.206	0.54041
768	1479.134	0.54137
769	1481.063	0.54214
770	1482.991	0.54272
771	1484.920	0.54321
772	1486.848	0.54355
773	1488.777	0.54383
774	1490.705	0.54436
775	1492.634	0.54486
776	1494.562	0.54511
777	1496.490	0.54516
778	1498.419	0.54538
779	1500.347	0.54576
780	1502.276	0.54596
781	1504.204	0.54590
782	1506.133	0.54593
783	1508.061	0.54636
784	1509.990	0.54693
785	1511.918	0.54712
786	1513.847	0.54712
787	1515.775	0.54716
788	1517.704	0.54724
789	1519.632	0.54731
790	1521.561	0.54723
791	1523.489	0.54716
792	1525.417	0.54727
793	1527.346	0.54745
794	1529.274	0.54764
795	1531.203	0.54772
796	1533.131	0.54767
797	1535.060	0.54760
798	1536.988	0.54742
799	1538.917	0.54713
800	1540.845	0.54706
801	1542.774	0.54722
802	1544.702	0.54751
803	1546.631	0.54775
804	1548.559	0.54778
805	1550.488	0.54763
806	1552.416	0.54751
807	1554.344	0.54749
808	1556.273	0.54728
809	1558.201	0.54689
810	1560.130	0.54691
811	1562.058	0.54733
812	1563.987	0.54776
813	1565.915	0.54796
814	1567.844	0.54791
815	1569.772	0.54777
816	1571.701	0.54775
817	1573.629	0.54765
818	1575.558	0.54749
819	1577.486	0.54769
820	1579.415	0.54799
821	1581.343	0.54797
822	1583.271	0.54783
823	1585.200	0.54775
824	1587.128	0.54768
825	1589.057	0.54770
826	1590.985	0.54774
827	1592.914	0.54783
828	1594.842	0.54796
829	1596.771	0.54802
830	1598.699	0.54811
831	1600.628	0.54810
832	1602.556	0.54803
833	1604.485	0.54807
834	1606.413	0.54806
835	1608.342	0.54807
836	1610.270	0.54818
837	1612.198	0.54825
838	1614.127	0.54812
839	1616.055	0.54778
840	1617.984	0.54777
841	1619.912	0.54803
842	1621.841	0.54793
843	1623.769	0.54778
844	1625.698	0.54782
845	1627.626	0.54773
846	1629.555	0.54778
847	1631.483	0.54789
848	1633.412	0.54779
849	1635.340	0.54763
850	1637.269	0.54773
851	1639.197	0.54803
852	1641.125	0.54829

853	1643.054	0.54829
854	1644.982	0.54800
855	1646.911	0.54780
856	1648.839	0.54798
857	1650.768	0.54786
858	1652.696	0.54740
859	1654.625	0.54763
860	1656.553	0.54828
861	1658.482	0.54848
862	1660.410	0.54832
863	1662.339	0.54822
864	1664.267	0.54837
865	1666.196	0.54854
866	1668.124	0.54841
867	1670.052	0.54837
868	1671.981	0.54849
869	1673.909	0.54849
870	1675.838	0.54847
871	1677.766	0.54847
872	1679.695	0.54857
873	1681.623	0.54852
874	1683.552	0.54817
875	1685.480	0.54820
876	1687.409	0.54872
877	1689.337	0.54904
878	1691.266	0.54901
879	1693.194	0.54859
880	1695.123	0.54807
881	1697.051	0.54788
882	1698.979	0.54784
883	1700.908	0.54785
884	1702.836	0.54796
885	1704.765	0.54794
886	1706.693	0.54771
887	1708.622	0.54750
888	1710.550	0.54735
889	1712.479	0.54720
890	1714.407	0.54678
891	1716.336	0.54627
892	1718.264	0.54618
893	1720.193	0.54638
894	1722.121	0.54645
895	1724.050	0.54651
896	1725.978	0.54661
897	1727.906	0.54661
898	1729.835	0.54668
899	1731.763	0.54674
900	1733.692	0.54678
901	1735.620	0.54716
902	1737.549	0.54764
903	1739.477	0.54783
904	1741.406	0.54813
905	1743.334	0.54877
906	1745.263	0.54921
907	1747.191	0.54921
908	1749.120	0.54908
909	1751.048	0.54917
910	1752.977	0.54955
911	1754.905	0.54982
912	1756.833	0.54995
913	1758.762	0.55011
914	1760.690	0.55009
915	1762.619	0.55010
916	1764.547	0.55031
917	1766.476	0.55048
918	1768.404	0.55056
919	1770.333	0.55045
920	1772.261	0.55038
921	1774.190	0.55045
922	1776.118	0.55048
923	1778.047	0.55045
924	1779.975	0.55044
925	1781.904	0.55036
926	1783.832	0.55029
927	1785.760	0.55039
928	1787.689	0.55065
929	1789.617	0.55068
930	1791.546	0.55038
931	1793.474	0.55027
932	1795.403	0.55043
933	1797.331	0.55054
934	1799.260	0.55052
935	1801.188	0.55049
936	1803.117	0.55059
937	1805.045	0.55066
938	1806.974	0.55061
939	1808.902	0.55066
940	1810.831	0.55076
941	1812.759	0.55076
942	1814.688	0.55080
943	1816.616	0.55087
944	1818.544	0.55087
945	1820.473	0.55077
946	1822.401	0.55072
947	1824.330	0.55067
948	1826.258	0.55066
949	1828.187	0.55076
950	1830.115	0.55075
951	1832.044	0.55077
952	1833.972	0.55095
953	1835.901	0.55093
954	1837.829	0.55072
955	1839.758	0.55077
956	1841.686	0.55085
957	1843.615	0.55094
958	1845.543	0.55106
959	1847.471	0.55111
960	1849.400	0.55111
961	1851.328	0.55108
962	1853.257	0.55104
963	1855.185	0.55102
964	1857.114	0.55106
965	1859.042	0.55107
966	1860.971	0.55102
967	1862.899	0.55091
968	1864.828	0.55092
969	1866.756	0.55103
970	1868.685	0.55119

971	1870.613	0.55120
972	1872.542	0.55120
973	1874.470	0.55130
974	1876.398	0.55128
975	1878.327	0.55127
976	1880.255	0.55126
977	1882.184	0.55114
978	1884.112	0.55115
979	1886.041	0.55135
980	1887.969	0.55145
981	1889.898	0.55144
982	1891.826	0.55151
983	1893.755	0.55160
984	1895.683	0.55151
985	1897.612	0.55143
986	1899.540	0.55144
987	1901.469	0.55142
988	1903.397	0.55136
989	1905.325	0.55132
990	1907.254	0.55138
991	1909.182	0.55157
992	1911.111	0.55171
993	1913.039	0.55173
994	1914.968	0.55166
995	1916.896	0.55157
996	1918.825	0.55150
997	1920.753	0.55146
998	1922.682	0.55149
999	1924.610	0.55163
1000	1926.539	0.55171
1001	1928.467	0.55177
1002	1930.396	0.55181
1003	1932.324	0.55170
1004	1934.252	0.55157
1005	1936.181	0.55158
1006	1938.109	0.55177
1007	1940.038	0.55189
1008	1941.966	0.55178
1009	1943.895	0.55170
1010	1945.823	0.55175
1011	1947.752	0.55175
1012	1949.680	0.55173
1013	1951.609	0.55173
1014	1953.537	0.55182
1015	1955.466	0.55193
1016	1957.394	0.55189
1017	1959.323	0.55190
1018	1961.251	0.55186
1019	1963.179	0.55184
1020	1965.108	0.55190
1021	1967.036	0.55204
1022	1968.965	0.55216
1023	1970.893	0.55219
1024	1972.822	0.55208
1025	1974.750	0.55200
1026	1976.679	0.55193
1027	1978.607	0.55197
1028	1980.536	0.55212
1029	1982.464	0.55225
1030	1984.393	0.55225
1031	1986.321	0.55205
1032	1988.250	0.55198
1033	1990.178	0.55215
1034	1992.106	0.55229
1035	1994.035	0.55237
1036	1995.963	0.55237
1037	1997.892	0.55225
1038	1999.820	0.55208
1039	2001.749	0.55200
1040	2003.677	0.55212
1041	2005.606	0.55222
1042	2007.534	0.55226
1043	2009.463	0.55236
1044	2011.391	0.55233
1045	2013.320	0.55229
1046	2015.248	0.55232
1047	2017.177	0.55236
1048	2019.105	0.55243
1049	2021.033	0.55238
1050	2022.962	0.55221
1051	2024.890	0.55218
1052	2026.819	0.55229
1053	2028.747	0.55240
1054	2030.676	0.55240
1055	2032.604	0.55233
1056	2034.533	0.55243
1057	2036.461	0.55231
1058	2038.390	0.55205
1059	2040.318	0.55204
1060	2042.247	0.55217
1061	2044.175	0.55235
1062	2046.104	0.55237
1063	2048.032	0.55238
1064	2049.960	0.55258
1065	2051.889	0.55270
1066	2053.817	0.55267
1067	2055.746	0.55268
1068	2057.674	0.55264
1069	2059.603	0.55263
1070	2061.531	0.55264
1071	2063.460	0.55269
1072	2065.388	0.55272
1073	2067.317	0.55266
1074	2069.245	0.55251
1075	2071.174	0.55247
1076	2073.102	0.55255
1077	2075.031	0.55255
1078	2076.959	0.55246
1079	2078.887	0.55254
1080	2080.816	0.55258
1081	2082.744	0.55245
1082	2084.673	0.55249
1083	2086.601	0.55274
1084	2088.530	0.55288
1085	2090.458	0.55287
1086	2092.387	0.55279
1087	2094.315	0.55282
1088	2096.244	0.55282

1089	2098.172	0.55283
1090	2100.101	0.55277
1091	2102.029	0.55254
1092	2103.958	0.55256
1093	2105.886	0.55267
1094	2107.814	0.55260
1095	2109.743	0.55263
1096	2111.671	0.55272
1097	2113.600	0.55272
1098	2115.528	0.55259
1099	2117.457	0.55248
1100	2119.385	0.55256

Sample: D26

800

301	578.540	0.45869
302	580.469	0.45896
303	582.397	0.45797
304	584.326	0.45554
305	586.254	0.45139
306	588.182	0.44603
307	590.111	0.43980
308	592.039	0.43209
309	593.968	0.42246
310	595.896	0.41030
311	597.825	0.39499
312	599.753	0.37663
313	601.682	0.35829
314	603.610	0.34327
315	605.539	0.33037
316	607.467	0.31856
317	609.396	0.31155
318	611.324	0.31199
319	613.253	0.31679
320	615.181	0.32038
321	617.109	0.32150
322	619.038	0.32284
323	620.966	0.32778
324	622.895	0.34335
325	624.823	0.37286
326	626.752	0.41071
327	628.680	0.44801
328	630.609	0.47592
329	632.537	0.49068
330	634.466	0.49610
331	636.394	0.49801
332	638.323	0.49958
333	640.251	0.50159
334	642.180	0.50326
335	644.108	0.50422
336	646.036	0.50476
337	647.965	0.50526
338	649.893	0.50586
339	651.822	0.50661
340	653.750	0.50694
341	655.679	0.50697
342	657.607	0.50728
343	659.536	0.50787
344	661.464	0.50897
345	663.393	0.50905
346	665.321	0.50599
347	667.250	0.50381
348	669.178	0.50704
349	671.107	0.51046
350	673.035	0.51128
351	674.963	0.51181
352	676.892	0.51190
353	678.820	0.51171
354	680.749	0.51190
355	682.677	0.51203
356	684.606	0.51183
357	686.534	0.51118
358	688.463	0.51088
359	690.391	0.51164
360	692.320	0.51219
361	694.248	0.51265
362	696.177	0.51454
363	698.105	0.51688
364	700.034	0.51756
365	701.962	0.51659
366	703.890	0.51525
367	705.819	0.51401
368	707.747	0.51192
369	709.676	0.50838
370	711.604	0.50306
371	713.533	0.49626
372	715.461	0.48957
373	717.390	0.48379
374	719.318	0.47910
375	721.247	0.47542
376	723.175	0.47155
377	725.104	0.46750
378	727.032	0.46400
379	728.961	0.46109
380	730.889	0.45875
381	732.817	0.45638
382	734.746	0.45334
383	736.674	0.45027
384	738.603	0.44855
385	740.531	0.45012
386	742.460	0.45404
387	744.388	0.45757
388	746.317	0.46058
389	748.245	0.46335
390	750.174	0.46540
391	752.102	0.46684
392	754.031	0.46799
393	755.959	0.46883
394	757.888	0.46950
395	759.816	0.47015
396	761.745	0.47121
397	763.673	0.47207
398	765.601	0.47256
399	767.530	0.47301
400	769.458	0.47314
401	771.387	0.47349
402	773.315	0.47389
403	775.244	0.47391
404	777.172	0.47437
405	779.101	0.47504
406	781.029	0.47507
407	782.958	0.47485
408	784.886	0.47478
409	786.815	0.47472
410	788.743	0.47499
411	790.672	0.47590
412	792.600	0.47718
413	794.528	0.47847
414	796.457	0.48078
415	798.385	0.48404
416	800.314	0.48584

417	802.242	0.48480
418	804.171	0.48233
419	806.099	0.48029
420	808.028	0.47874
421	809.956	0.47701
422	811.885	0.47560
423	813.813	0.47444
424	815.742	0.47334
425	817.670	0.47305
426	819.599	0.47359
427	821.527	0.47490
428	823.455	0.47693
429	825.384	0.47906
430	827.312	0.48151
431	829.241	0.48453
432	831.169	0.48756
433	833.098	0.48988
434	835.026	0.49121
435	836.955	0.49186
436	838.883	0.49216
437	840.812	0.49206
438	842.740	0.49143
439	844.669	0.49016
440	846.597	0.48903
441	848.526	0.48813
442	850.454	0.48692
443	852.382	0.48569
444	854.311	0.48456
445	856.239	0.48308
446	858.168	0.48118
447	860.096	0.47914
448	862.025	0.47742
449	863.953	0.47584
450	865.882	0.47398
451	867.810	0.47234
452	869.739	0.47112
453	871.667	0.47040
454	873.596	0.47002
455	875.524	0.46967
456	877.453	0.46921
457	879.381	0.46846
458	881.309	0.46768
459	883.238	0.46736
460	885.166	0.46704
461	887.095	0.46672
462	889.023	0.46660
463	890.952	0.46678
464	892.880	0.46729
465	894.809	0.46755
466	896.737	0.46781
467	898.666	0.46822
468	900.594	0.46837
469	902.523	0.46853
470	904.451	0.46930
471	906.380	0.47070
472	908.308	0.47223
473	910.236	0.47340
474	912.165	0.47481
475	914.093	0.47670
476	916.022	0.47843
477	917.950	0.48000
478	919.879	0.48138
479	921.807	0.48247
480	923.736	0.48321
481	925.664	0.48392
482	927.593	0.48513
483	929.521	0.48637
484	931.450	0.48736
485	933.378	0.48817
486	935.307	0.48846
487	937.235	0.48885
488	939.163	0.48963
489	941.092	0.49026
490	943.020	0.49037
491	944.949	0.49000
492	946.877	0.48963
493	948.806	0.48965
494	950.734	0.48952
495	952.663	0.48911
496	954.591	0.48879
497	956.520	0.48839
498	958.448	0.48811
499	960.377	0.48808
500	962.305	0.48831
501	964.234	0.48846
502	966.162	0.48840
503	968.090	0.48879
504	970.019	0.48958
505	971.947	0.49042
506	973.876	0.49126
507	975.804	0.49199
508	977.733	0.49276
509	979.661	0.49373
510	981.590	0.49483
511	983.518	0.49620
512	985.447	0.49746
513	987.375	0.49862
514	989.304	0.49991
515	991.232	0.50110
516	993.161	0.50213
517	995.089	0.50309
518	997.017	0.50410
519	998.946	0.50528
520	1000.874	0.50620
521	1002.803	0.50668
522	1004.731	0.50716
523	1006.660	0.50799
524	1008.588	0.50879
525	1010.517	0.50912
526	1012.445	0.50938
527	1014.374	0.50998
528	1016.302	0.51060
529	1018.231	0.51135
530	1020.159	0.51226
531	1022.088	0.51275
532	1024.016	0.51306
533	1025.944	0.51358
534	1027.873	0.51414

535	1029.801	0.51438
536	1031.730	0.51450
537	1033.658	0.51490
538	1035.587	0.51519
539	1037.515	0.51514
540	1039.444	0.51509
541	1041.372	0.51504
542	1043.301	0.51500
543	1045.229	0.51496
544	1047.158	0.51463
545	1049.086	0.51410
546	1051.015	0.51349
547	1052.943	0.51296
548	1054.871	0.51269
549	1056.800	0.51238
550	1058.728	0.51167
551	1060.657	0.51059
552	1062.585	0.50921
553	1064.514	0.50798
554	1066.442	0.50687
555	1068.371	0.50564
556	1070.299	0.50397
557	1072.228	0.50173
558	1074.156	0.49890
559	1076.085	0.49548
560	1078.013	0.49149
561	1079.942	0.48683
562	1081.870	0.48161
563	1083.798	0.47565
564	1085.727	0.46892
565	1087.655	0.46169
566	1089.584	0.45373
567	1091.512	0.44506
568	1093.441	0.43641
569	1095.369	0.42777
570	1097.298	0.41940
571	1099.226	0.41216
572	1101.155	0.40614
573	1103.083	0.40174
574	1105.012	0.39901
575	1106.940	0.39753
576	1108.869	0.39778
577	1110.797	0.39984
578	1112.725	0.40316
579	1114.654	0.40820
580	1116.582	0.41477
581	1118.511	0.42246
582	1120.439	0.43097
583	1122.368	0.44007
584	1124.296	0.44954
585	1126.225	0.45902
586	1128.153	0.46810
587	1130.082	0.47617
588	1132.010	0.48266
589	1133.939	0.48787
590	1135.867	0.49200
591	1137.796	0.49531
592	1139.724	0.49815
593	1141.652	0.50017
594	1143.581	0.50171
595	1145.509	0.50321
596	1147.438	0.50446
597	1149.366	0.50551
598	1151.295	0.50633
599	1153.223	0.50694
600	1155.152	0.50774
601	1157.080	0.50854
602	1159.009	0.50921
603	1160.937	0.50972
604	1162.866	0.51017
605	1164.794	0.51050
606	1166.723	0.51077
607	1168.651	0.51137
608	1170.579	0.51212
609	1172.508	0.51265
610	1174.436	0.51316
611	1176.365	0.51377
612	1178.293	0.51451
613	1180.222	0.51507
614	1182.150	0.51524
615	1184.079	0.51555
616	1186.007	0.51606
617	1187.936	0.51664
618	1189.864	0.51739
619	1191.793	0.51803
620	1193.721	0.51867
621	1195.650	0.51942
622	1197.578	0.51988
623	1199.506	0.52024
624	1201.435	0.52062
625	1203.363	0.52095
626	1205.292	0.52146
627	1207.220	0.52195
628	1209.149	0.52234
629	1211.077	0.52264
630	1213.006	0.52283
631	1214.934	0.52291
632	1216.863	0.52285
633	1218.791	0.52297
634	1220.720	0.52304
635	1222.648	0.52285
636	1224.577	0.52286
637	1226.505	0.52303
638	1228.433	0.52305
639	1230.362	0.52296
640	1232.290	0.52288
641	1234.219	0.52302
642	1236.147	0.52324
643	1238.076	0.52335
644	1240.004	0.52343
645	1241.933	0.52333
646	1243.861	0.52329
647	1245.790	0.52345
648	1247.718	0.52362
649	1249.647	0.52381
650	1251.575	0.52394
651	1253.504	0.52398
652	1255.432	0.52395

653	1257.360	0.52371
654	1259.289	0.52354
655	1261.217	0.52362
656	1263.146	0.52379
657	1265.074	0.52394
658	1267.003	0.52391
659	1268.931	0.52383
660	1270.860	0.52366
661	1272.788	0.52333
662	1274.717	0.52307
663	1276.645	0.52292
664	1278.574	0.52281
665	1280.502	0.52264
666	1282.431	0.52238
667	1284.359	0.52213
668	1286.287	0.52179
669	1288.216	0.52135
670	1290.144	0.52116
671	1292.073	0.52106
672	1294.001	0.52104
673	1295.930	0.52106
674	1297.858	0.52111
675	1299.787	0.52130
676	1301.715	0.52145
677	1303.644	0.52148
678	1305.572	0.52154
679	1307.501	0.52165
680	1309.429	0.52202
681	1311.358	0.52238
682	1313.286	0.52274
683	1315.214	0.52331
684	1317.143	0.52372
685	1319.071	0.52390
686	1321.000	0.52410
687	1322.928	0.52449
688	1324.857	0.52510
689	1326.785	0.52575
690	1328.714	0.52624
691	1330.642	0.52647
692	1332.571	0.52665
693	1334.499	0.52681
694	1336.428	0.52688
695	1338.356	0.52709
696	1340.285	0.52731
697	1342.213	0.52727
698	1344.141	0.52728
699	1346.070	0.52726
700	1347.998	0.52719
701	1349.927	0.52716
702	1351.855	0.52698
703	1353.784	0.52687
704	1355.712	0.52695
705	1357.641	0.52678
706	1359.569	0.52647
707	1361.498	0.52620
708	1363.426	0.52605
709	1365.355	0.52601
710	1367.283	0.52595
711	1369.212	0.52599
712	1371.140	0.52592
713	1373.068	0.52565
714	1374.997	0.52567
715	1376.925	0.52588
716	1378.854	0.52622
717	1380.782	0.52655
718	1382.711	0.52661
719	1384.639	0.52639
720	1386.568	0.52600
721	1388.496	0.52584
722	1390.425	0.52612
723	1392.353	0.52635
724	1394.282	0.52636
725	1396.210	0.52639
726	1398.139	0.52651
727	1400.067	0.52668
728	1401.995	0.52694
729	1403.924	0.52709
730	1405.852	0.52680
731	1407.781	0.52642
732	1409.709	0.52648
733	1411.638	0.52655
734	1413.566	0.52643
735	1415.495	0.52614
736	1417.423	0.52548
737	1419.352	0.52501
738	1421.280	0.52497
739	1423.209	0.52489
740	1425.137	0.52474
741	1427.066	0.52445
742	1428.994	0.52392
743	1430.922	0.52333
744	1432.851	0.52268
745	1434.779	0.52209
746	1436.708	0.52167
747	1438.636	0.52149
748	1440.565	0.52132
749	1442.493	0.52089
750	1444.422	0.52060
751	1446.350	0.52067
752	1448.279	0.52074
753	1450.207	0.52086
754	1452.136	0.52111
755	1454.064	0.52118
756	1455.993	0.52111
757	1457.921	0.52166
758	1459.849	0.52286
759	1461.778	0.52385
760	1463.706	0.52464
761	1465.635	0.52553
762	1467.563	0.52638
763	1469.492	0.52700
764	1471.420	0.52749
765	1473.349	0.52822
766	1475.277	0.52932
767	1477.206	0.53042
768	1479.134	0.53138
769	1481.063	0.53213
770	1482.991	0.53260

771	1484.920	0.53306
772	1486.848	0.53346
773	1488.777	0.53368
774	1490.705	0.53391
775	1492.634	0.53433
776	1494.562	0.53475
777	1496.490	0.53485
778	1498.419	0.53506
779	1500.347	0.53564
780	1502.276	0.53604
781	1504.204	0.53576
782	1506.133	0.53531
783	1508.061	0.53562
784	1509.990	0.53630
785	1511.918	0.53653
786	1513.847	0.53662
787	1515.775	0.53660
788	1517.704	0.53642
789	1519.632	0.53641
790	1521.561	0.53652
791	1523.489	0.53668
792	1525.417	0.53681
793	1527.346	0.53684
794	1529.274	0.53715
795	1531.203	0.53733
796	1533.131	0.53714
797	1535.060	0.53711
798	1536.988	0.53691
799	1538.917	0.53637
800	1540.845	0.53616
801	1542.774	0.53636
802	1544.702	0.53685
803	1546.631	0.53734
804	1548.559	0.53745
805	1550.488	0.53726
806	1552.416	0.53712
807	1554.344	0.53708
808	1556.273	0.53672
809	1558.201	0.53598
810	1560.130	0.53600
811	1562.058	0.53693
812	1563.987	0.53751
813	1565.915	0.53757
814	1567.844	0.53751
815	1569.772	0.53747
816	1571.701	0.53761
817	1573.629	0.53754
818	1575.558	0.53700
819	1577.486	0.53696
820	1579.415	0.53750
821	1581.343	0.53771
822	1583.271	0.53769
823	1585.200	0.53788
824	1587.128	0.53806
825	1589.057	0.53810
826	1590.985	0.53796
827	1592.914	0.53778
828	1594.842	0.53780
829	1596.771	0.53787
830	1598.699	0.53802
831	1600.628	0.53826
832	1602.556	0.53817
833	1604.485	0.53794
834	1606.413	0.53780
835	1608.342	0.53773
836	1610.270	0.53776
837	1612.198	0.53795
838	1614.127	0.53795
839	1616.055	0.53757
840	1617.984	0.53756
841	1619.912	0.53801
842	1621.841	0.53817
843	1623.769	0.53808
844	1625.698	0.53803
845	1627.626	0.53798
846	1629.555	0.53822
847	1631.483	0.53837
848	1633.412	0.53804
849	1635.340	0.53768
850	1637.269	0.53785
851	1639.197	0.53827
852	1641.125	0.53853
853	1643.054	0.53850
854	1644.982	0.53818
855	1646.911	0.53791
856	1648.839	0.53798
857	1650.768	0.53769
858	1652.696	0.53709
859	1654.625	0.53744
860	1656.553	0.53832
861	1658.482	0.53868
862	1660.410	0.53860
863	1662.339	0.53841
864	1664.267	0.53858
865	1666.196	0.53879
866	1668.124	0.53856
867	1670.052	0.53849
868	1671.981	0.53858
869	1673.909	0.53847
870	1675.838	0.53863
871	1677.766	0.53897
872	1679.695	0.53915
873	1681.623	0.53881
874	1683.552	0.53820
875	1685.480	0.53828
876	1687.409	0.53885
877	1689.337	0.53916
878	1691.266	0.53930
879	1693.194	0.53895
880	1695.123	0.53833
881	1697.051	0.53801
882	1698.979	0.53778
883	1700.908	0.53766
884	1702.836	0.53786
885	1704.765	0.53797
886	1706.693	0.53785
887	1708.622	0.53773
888	1710.550	0.53755

889	1712.479	0.53733
890	1714.407	0.53689
891	1716.336	0.53626
892	1718.264	0.53601
893	1720.193	0.53629
894	1722.121	0.53662
895	1724.050	0.53680
896	1725.978	0.53694
897	1727.906	0.53704
898	1729.835	0.53717
899	1731.763	0.53695
900	1733.692	0.53677
901	1735.620	0.53722
902	1737.549	0.53777
903	1739.477	0.53802
904	1741.406	0.53841
905	1743.334	0.53905
906	1745.263	0.53955
907	1747.191	0.53969
908	1749.120	0.53966
909	1751.048	0.53975
910	1752.977	0.54018
911	1754.905	0.54055
912	1756.833	0.54069
913	1758.762	0.54078
914	1760.690	0.54084
915	1762.619	0.54087
916	1764.547	0.54104
917	1766.476	0.54107
918	1768.404	0.54105
919	1770.333	0.54099
920	1772.261	0.54086
921	1774.190	0.54090
922	1776.118	0.54103
923	1778.047	0.54104
924	1779.975	0.54125
925	1781.904	0.54144
926	1783.832	0.54139
927	1785.760	0.54130
928	1787.689	0.54133
929	1789.617	0.54119
930	1791.546	0.54094
931	1793.474	0.54083
932	1795.403	0.54111
933	1797.331	0.54140
934	1799.260	0.54149
935	1801.188	0.54140
936	1803.117	0.54131
937	1805.045	0.54120
938	1806.974	0.54131
939	1808.902	0.54158
940	1810.831	0.54170
941	1812.759	0.54170
942	1814.688	0.54183
943	1816.616	0.54190
944	1818.544	0.54189
945	1820.473	0.54184
946	1822.401	0.54170
947	1824.330	0.54165
948	1826.258	0.54164
949	1828.187	0.54178
950	1830.115	0.54190
951	1832.044	0.54186
952	1833.972	0.54198
953	1835.901	0.54199
954	1837.829	0.54183
955	1839.758	0.54195
956	1841.686	0.54197
957	1843.615	0.54189
958	1845.543	0.54199
959	1847.471	0.54213
960	1849.400	0.54226
961	1851.328	0.54232
962	1853.257	0.54223
963	1855.185	0.54226
964	1857.114	0.54229
965	1859.042	0.54225
966	1860.971	0.54228
967	1862.899	0.54240
968	1864.828	0.54246
969	1866.756	0.54227
970	1868.685	0.54211
971	1870.613	0.54228
972	1872.542	0.54245
973	1874.470	0.54257
974	1876.398	0.54266
975	1878.327	0.54264
976	1880.255	0.54269
977	1882.184	0.54274
978	1884.112	0.54275
979	1886.041	0.54280
980	1887.969	0.54275
981	1889.898	0.54270
982	1891.826	0.54272
983	1893.755	0.54270
984	1895.683	0.54266
985	1897.612	0.54270
986	1899.540	0.54280
987	1901.469	0.54296
988	1903.397	0.54310
989	1905.325	0.54299
990	1907.254	0.54289
991	1909.182	0.54299
992	1911.111	0.54304
993	1913.039	0.54312
994	1914.968	0.54317
995	1916.896	0.54300
996	1918.825	0.54299
997	1920.753	0.54309
998	1922.682	0.54300
999	1924.610	0.54306
1000	1926.539	0.54322
1001	1928.467	0.54343
1002	1930.396	0.54356
1003	1932.324	0.54341
1004	1934.252	0.54343
1005	1936.181	0.54364
1006	1938.109	0.54371

1007	1940.038	0.54362
1008	1941.966	0.54346
1009	1943.895	0.54339
1010	1945.823	0.54358
1011	1947.752	0.54372
1012	1949.680	0.54372
1013	1951.609	0.54361
1014	1953.537	0.54363
1015	1955.466	0.54377
1016	1957.394	0.54389
1017	1959.323	0.54403
1018	1961.251	0.54409
1019	1963.179	0.54400
1020	1965.108	0.54393
1021	1967.036	0.54404
1022	1968.965	0.54424
1023	1970.893	0.54420
1024	1972.822	0.54405
1025	1974.750	0.54418
1026	1976.679	0.54425
1027	1978.607	0.54412
1028	1980.536	0.54398
1029	1982.464	0.54404
1030	1984.393	0.54415
1031	1986.321	0.54405
1032	1988.250	0.54416
1033	1990.178	0.54436
1034	1992.106	0.54432
1035	1994.035	0.54434
1036	1995.963	0.54429
1037	1997.892	0.54423
1038	1999.820	0.54440
1039	2001.749	0.54448
1040	2003.677	0.54446
1041	2005.606	0.54442
1042	2007.534	0.54438
1043	2009.463	0.54461
1044	2011.391	0.54474
1045	2013.320	0.54467
1046	2015.248	0.54462
1047	2017.177	0.54457
1048	2019.105	0.54450
1049	2021.033	0.54454
1050	2022.962	0.54460
1051	2024.890	0.54462
1052	2026.819	0.54467
1053	2028.747	0.54476
1054	2030.676	0.54504
1055	2032.604	0.54498
1056	2034.533	0.54464
1057	2036.461	0.54448
1058	2038.390	0.54466
1059	2040.318	0.54483
1060	2042.247	0.54491
1061	2044.175	0.54496
1062	2046.104	0.54504
1063	2048.032	0.54510
1064	2049.960	0.54503
1065	2051.889	0.54498
1066	2053.817	0.54494
1067	2055.746	0.54487
1068	2057.674	0.54500
1069	2059.603	0.54503
1070	2061.531	0.54494
1071	2063.460	0.54508
1072	2065.388	0.54506
1073	2067.317	0.54492
1074	2069.245	0.54494
1075	2071.174	0.54493
1076	2073.102	0.54494
1077	2075.031	0.54493
1078	2076.959	0.54489
1079	2078.887	0.54504
1080	2080.816	0.54517
1081	2082.744	0.54519
1082	2084.673	0.54529
1083	2086.601	0.54532
1084	2088.530	0.54538
1085	2090.458	0.54548
1086	2092.387	0.54541
1087	2094.315	0.54523
1088	2096.244	0.54503
1089	2098.172	0.54497
1090	2100.101	0.54511
1091	2102.029	0.54507
1092	2103.958	0.54492
1093	2105.886	0.54503
1094	2107.814	0.54530
1095	2109.743	0.54544
1096	2111.671	0.54538
1097	2113.600	0.54524
1098	2115.528	0.54524
1099	2117.457	0.54537
1100	2119.385	0.54540

Sample: D36

801

300	576.612	0.48351
301	578.540	0.48454
302	580.469	0.48465
303	582.397	0.48359
304	584.326	0.48132
305	586.254	0.47797
306	588.182	0.47333
307	590.111	0.46655
308	592.039	0.45803
309	593.968	0.44822
310	595.896	0.43689
311	597.825	0.42363
312	599.753	0.40745
313	601.682	0.38996
314	603.610	0.37484
315	605.539	0.36203
316	607.467	0.35046
317	609.396	0.34313
318	611.324	0.34290
319	613.253	0.34759
320	615.181	0.35153
321	617.109	0.35291
322	619.038	0.35417
323	620.966	0.35908
324	622.895	0.37410
325	624.823	0.40217
326	626.752	0.43800
327	628.680	0.47309
328	630.609	0.49883
329	632.537	0.51246
330	634.466	0.51778
331	636.394	0.51938
332	638.323	0.52025
333	640.251	0.52141
334	642.180	0.52253
335	644.108	0.52388
336	646.036	0.52506
337	647.965	0.52597
338	649.893	0.52667
339	651.822	0.52671
340	653.750	0.52638
341	655.679	0.52658
342	657.607	0.52739
343	659.536	0.52798
344	661.464	0.52805
345	663.393	0.52846
346	665.321	0.52779
347	667.250	0.52602
348	669.178	0.52676
349	671.107	0.52896
350	673.035	0.52998
351	674.963	0.53081
352	676.892	0.53103
353	678.820	0.53062
354	680.749	0.53046
355	682.677	0.53054
356	684.606	0.53090
357	686.534	0.53106
358	688.463	0.53072
359	690.391	0.53056
360	692.320	0.53096
361	694.248	0.53248
362	696.177	0.53445
363	698.105	0.53526
364	700.034	0.53473
365	701.962	0.53382
366	703.890	0.53273
367	705.819	0.53114
368	707.747	0.52866
369	709.676	0.52567
370	711.604	0.52189
371	713.533	0.51672
372	715.461	0.51033
373	717.390	0.50437
374	719.318	0.49988
375	721.247	0.49641
376	723.175	0.49334
377	725.104	0.49008
378	727.032	0.48621
379	728.961	0.48300
380	730.889	0.48065
381	732.817	0.47798
382	734.746	0.47517
383	736.674	0.47337
384	738.603	0.47292
385	740.531	0.47423
386	742.460	0.47717
387	744.388	0.48020
388	746.317	0.48250
389	748.245	0.48485
390	750.174	0.48674
391	752.102	0.48744
392	754.031	0.48834
393	755.959	0.48963
394	757.888	0.49069
395	759.816	0.49134
396	761.745	0.49142
397	763.673	0.49158
398	765.601	0.49202
399	767.530	0.49259
400	769.458	0.49346
401	771.387	0.49405
402	773.315	0.49398
403	775.244	0.49413
404	777.172	0.49463
405	779.101	0.49528
406	781.029	0.49534
407	782.958	0.49496
408	784.886	0.49512
409	786.815	0.49562
410	788.743	0.49614
411	790.672	0.49678
412	792.600	0.49718
413	794.528	0.49800
414	796.457	0.50035
415	798.385	0.50307

416	800.314	0.50437
417	802.242	0.50355
418	804.171	0.50134
419	806.099	0.49916
420	808.028	0.49767
421	809.956	0.49654
422	811.885	0.49526
423	813.813	0.49371
424	815.742	0.49271
425	817.670	0.49298
426	819.599	0.49369
427	821.527	0.49464
428	823.455	0.49608
429	825.384	0.49781
430	827.312	0.50033
431	829.241	0.50364
432	831.169	0.50648
433	833.098	0.50822
434	835.026	0.50935
435	836.955	0.51036
436	838.883	0.51091
437	840.812	0.51042
438	842.740	0.50917
439	844.669	0.50767
440	846.597	0.50684
441	848.526	0.50656
442	850.454	0.50584
443	852.382	0.50472
444	854.311	0.50316
445	856.239	0.50129
446	858.168	0.49961
447	860.096	0.49807
448	862.025	0.49654
449	863.953	0.49515
450	865.882	0.49382
451	867.810	0.49260
452	869.739	0.49112
453	871.667	0.48969
454	873.596	0.48904
455	875.524	0.48872
456	877.453	0.48846
457	879.381	0.48816
458	881.309	0.48722
459	883.238	0.48621
460	885.166	0.48564
461	887.095	0.48559
462	889.023	0.48598
463	890.952	0.48613
464	892.880	0.48586
465	894.809	0.48573
466	896.737	0.48599
467	898.666	0.48632
468	900.594	0.48660
469	902.523	0.48719
470	904.451	0.48814
471	906.380	0.48912
472	908.308	0.49006
473	910.236	0.49115
474	912.165	0.49262
475	914.093	0.49439
476	916.022	0.49600
477	917.950	0.49722
478	919.879	0.49816
479	921.807	0.49902
480	923.736	0.50007
481	925.664	0.50112
482	927.593	0.50181
483	929.521	0.50223
484	931.450	0.50318
485	933.378	0.50432
486	935.307	0.50479
487	937.235	0.50479
488	939.163	0.50486
489	941.092	0.50478
490	943.020	0.50490
491	944.949	0.50520
492	946.877	0.50494
493	948.806	0.50436
494	950.734	0.50414
495	952.663	0.50382
496	954.591	0.50336
497	956.520	0.50330
498	958.448	0.50335
499	960.377	0.50313
500	962.305	0.50308
501	964.234	0.50332
502	966.162	0.50369
503	968.090	0.50406
504	970.019	0.50426
505	971.947	0.50483
506	973.876	0.50567
507	975.804	0.50634
508	977.733	0.50669
509	979.661	0.50717
510	981.590	0.50824
511	983.518	0.50965
512	985.447	0.51071
513	987.375	0.51160
514	989.304	0.51285
515	991.232	0.51403
516	993.161	0.51492
517	995.089	0.51600
518	997.017	0.51698
519	998.946	0.51746
520	1000.874	0.51797
521	1002.803	0.51880
522	1004.731	0.51950
523	1006.660	0.52009
524	1008.588	0.52063
525	1010.517	0.52095
526	1012.445	0.52132
527	1014.374	0.52201
528	1016.302	0.52285
529	1018.231	0.52341
530	1020.159	0.52390
531	1022.088	0.52452
532	1024.016	0.52498
533	1025.944	0.52516

534	1027.873	0.52518
535	1029.801	0.52540
536	1031.730	0.52587
537	1033.658	0.52609
538	1035.587	0.52612
539	1037.515	0.52628
540	1039.444	0.52635
541	1041.372	0.52621
542	1043.301	0.52631
543	1045.229	0.52643
544	1047.158	0.52620
545	1049.086	0.52571
546	1051.015	0.52517
547	1052.943	0.52468
548	1054.871	0.52444
549	1056.800	0.52415
550	1058.728	0.52343
551	1060.657	0.52285
552	1062.585	0.52246
553	1064.514	0.52189
554	1066.442	0.52089
555	1068.371	0.51945
556	1070.299	0.51772
557	1072.228	0.51602
558	1074.156	0.51408
559	1076.085	0.51152
560	1078.013	0.50849
561	1079.942	0.50512
562	1081.870	0.50132
563	1083.798	0.49683
564	1085.727	0.49164
565	1087.655	0.48592
566	1089.584	0.47975
567	1091.512	0.47319
568	1093.441	0.46629
569	1095.369	0.45937
570	1097.298	0.45295
571	1099.226	0.44741
572	1101.155	0.44279
573	1103.083	0.43904
574	1105.012	0.43652
575	1106.940	0.43551
576	1108.869	0.43583
577	1110.797	0.43735
578	1112.725	0.44005
579	1114.654	0.44397
580	1116.582	0.44884
581	1118.511	0.45455
582	1120.439	0.46110
583	1122.368	0.46834
584	1124.296	0.47592
585	1126.225	0.48342
586	1128.153	0.49029
587	1130.082	0.49619
588	1132.010	0.50088
589	1133.939	0.50470
590	1135.867	0.50813
591	1137.796	0.51083
592	1139.724	0.51270
593	1141.652	0.51418
594	1143.581	0.51526
595	1145.509	0.51606
596	1147.438	0.51706
597	1149.366	0.51817
598	1151.295	0.51904
599	1153.223	0.51973
600	1155.152	0.52035
601	1157.080	0.52082
602	1159.009	0.52126
603	1160.937	0.52167
604	1162.866	0.52203
605	1164.794	0.52251
606	1166.723	0.52308
607	1168.651	0.52351
608	1170.579	0.52398
609	1172.508	0.52460
610	1174.436	0.52520
611	1176.365	0.52574
612	1178.293	0.52626
613	1180.222	0.52668
614	1182.150	0.52697
615	1184.079	0.52733
616	1186.007	0.52780
617	1187.936	0.52838
618	1189.864	0.52894
619	1191.793	0.52945
620	1193.721	0.52987
621	1195.650	0.53042
622	1197.578	0.53109
623	1199.506	0.53160
624	1201.435	0.53196
625	1203.363	0.53234
626	1205.292	0.53265
627	1207.220	0.53298
628	1209.149	0.53347
629	1211.077	0.53381
630	1213.006	0.53397
631	1214.934	0.53408
632	1216.863	0.53400
633	1218.791	0.53387
634	1220.720	0.53373
635	1222.648	0.53371
636	1224.577	0.53388
637	1226.505	0.53404
638	1228.433	0.53408
639	1230.362	0.53415
640	1232.290	0.53412
641	1234.219	0.53417
642	1236.147	0.53437
643	1238.076	0.53458
644	1240.004	0.53470
645	1241.933	0.53472
646	1243.861	0.53471
647	1245.790	0.53467
648	1247.718	0.53466
649	1249.647	0.53465
650	1251.575	0.53475
651	1253.504	0.53495

652	1255.432	0.53518
653	1257.360	0.53542
654	1259.289	0.53552
655	1261.217	0.53544
656	1263.146	0.53554
657	1265.074	0.53559
658	1267.003	0.53536
659	1268.931	0.53529
660	1270.860	0.53534
661	1272.788	0.53518
662	1274.717	0.53498
663	1276.645	0.53473
664	1278.574	0.53434
665	1280.502	0.53410
666	1282.431	0.53416
667	1284.359	0.53401
668	1286.287	0.53365
669	1288.216	0.53350
670	1290.144	0.53340
671	1292.073	0.53336
672	1294.001	0.53346
673	1295.930	0.53331
674	1297.858	0.53298
675	1299.787	0.53303
676	1301.715	0.53321
677	1303.644	0.53332
678	1305.572	0.53354
679	1307.501	0.53386
680	1309.429	0.53413
681	1311.358	0.53435
682	1313.286	0.53465
683	1315.214	0.53489
684	1317.143	0.53507
685	1319.071	0.53538
686	1321.000	0.53586
687	1322.928	0.53639
688	1324.857	0.53683
689	1326.785	0.53716
690	1328.714	0.53763
691	1330.642	0.53817
692	1332.571	0.53838
693	1334.499	0.53823
694	1336.428	0.53825
695	1338.356	0.53854
696	1340.285	0.53869
697	1342.213	0.53875
698	1344.141	0.53881
699	1346.070	0.53871
700	1347.998	0.53869
701	1349.927	0.53878
702	1351.855	0.53871
703	1353.784	0.53855
704	1355.712	0.53850
705	1357.641	0.53819
706	1359.569	0.53768
707	1361.498	0.53768
708	1363.426	0.53793
709	1365.355	0.53775
710	1367.283	0.53762
711	1369.212	0.53777
712	1371.140	0.53765
713	1373.068	0.53753
714	1374.997	0.53769
715	1376.925	0.53784
716	1378.854	0.53802
717	1380.782	0.53828
718	1382.711	0.53815
719	1384.639	0.53763
720	1386.568	0.53736
721	1388.496	0.53757
722	1390.425	0.53780
723	1392.353	0.53803
724	1394.282	0.53827
725	1396.210	0.53831
726	1398.139	0.53827
727	1400.067	0.53834
728	1401.995	0.53845
729	1403.924	0.53847
730	1405.852	0.53841
731	1407.781	0.53835
732	1409.709	0.53839
733	1411.638	0.53851
734	1413.566	0.53834
735	1415.495	0.53787
736	1417.423	0.53733
737	1419.352	0.53689
738	1421.280	0.53681
739	1423.209	0.53695
740	1425.137	0.53672
741	1427.066	0.53625
742	1428.994	0.53590
743	1430.922	0.53540
744	1432.851	0.53473
745	1434.779	0.53419
746	1436.708	0.53392
747	1438.636	0.53366
748	1440.565	0.53327
749	1442.493	0.53295
750	1444.422	0.53270
751	1446.350	0.53273
752	1448.279	0.53303
753	1450.207	0.53308
754	1452.136	0.53306
755	1454.064	0.53328
756	1455.993	0.53345
757	1457.921	0.53379
758	1459.849	0.53466
759	1461.778	0.53555
760	1463.706	0.53612
761	1465.635	0.53688
762	1467.563	0.53795
763	1469.492	0.53861
764	1471.420	0.53885
765	1473.349	0.53953
766	1475.277	0.54058
767	1477.206	0.54152
768	1479.134	0.54231
769	1481.063	0.54297

770	1482.991	0.54361
771	1484.920	0.54424
772	1486.848	0.54453
773	1488.777	0.54460
774	1490.705	0.54498
775	1492.634	0.54552
776	1494.562	0.54572
777	1496.490	0.54573
778	1498.419	0.54594
779	1500.347	0.54626
780	1502.276	0.54657
781	1504.204	0.54634
782	1506.133	0.54576
783	1508.061	0.54604
784	1509.990	0.54688
785	1511.918	0.54716
786	1513.847	0.54719
787	1515.775	0.54741
788	1517.704	0.54738
789	1519.632	0.54716
790	1521.561	0.54714
791	1523.489	0.54714
792	1525.417	0.54700
793	1527.346	0.54718
794	1529.274	0.54773
795	1531.203	0.54781
796	1533.131	0.54767
797	1535.060	0.54786
798	1536.988	0.54741
799	1538.917	0.54646
800	1540.845	0.54636
801	1542.774	0.54688
802	1544.702	0.54746
803	1546.631	0.54801
804	1548.559	0.54792
805	1550.488	0.54749
806	1552.416	0.54755
807	1554.344	0.54771
808	1556.273	0.54722
809	1558.201	0.54650
810	1560.130	0.54655
811	1562.058	0.54714
812	1563.987	0.54752
813	1565.915	0.54780
814	1567.844	0.54783
815	1569.772	0.54774
816	1571.701	0.54798
817	1573.629	0.54789
818	1575.558	0.54747
819	1577.486	0.54766
820	1579.415	0.54792
821	1581.343	0.54784
822	1583.271	0.54796
823	1585.200	0.54811
824	1587.128	0.54802
825	1589.057	0.54807
826	1590.985	0.54824
827	1592.914	0.54819
828	1594.842	0.54814
829	1596.771	0.54824
830	1598.699	0.54819
831	1600.628	0.54819
832	1602.556	0.54825
833	1604.485	0.54801
834	1606.413	0.54791
835	1608.342	0.54815
836	1610.270	0.54814
837	1612.198	0.54804
838	1614.127	0.54800
839	1616.055	0.54778
840	1617.984	0.54787
841	1619.912	0.54834
842	1621.841	0.54836
843	1623.769	0.54795
844	1625.698	0.54782
845	1627.626	0.54800
846	1629.555	0.54808
847	1631.483	0.54807
848	1633.412	0.54792
849	1635.340	0.54764
850	1637.269	0.54761
851	1639.197	0.54801
852	1641.125	0.54824
853	1643.054	0.54807
854	1644.982	0.54789
855	1646.911	0.54794
856	1648.839	0.54800
857	1650.768	0.54757
858	1652.696	0.54695
859	1654.625	0.54730
860	1656.553	0.54830
861	1658.482	0.54869
862	1660.410	0.54853
863	1662.339	0.54840
864	1664.267	0.54858
865	1666.196	0.54863
866	1668.124	0.54823
867	1670.052	0.54810
868	1671.981	0.54835
869	1673.909	0.54834
870	1675.838	0.54839
871	1677.766	0.54859
872	1679.695	0.54872
873	1681.623	0.54862
874	1683.552	0.54808
875	1685.480	0.54803
876	1687.409	0.54872
877	1689.337	0.54897
878	1691.266	0.54870
879	1693.194	0.54834
880	1695.123	0.54791
881	1697.051	0.54762
882	1698.979	0.54734
883	1700.908	0.54724
884	1702.836	0.54752
885	1704.765	0.54772
886	1706.693	0.54778
887	1708.622	0.54774

888	1710.550	0.54740
889	1712.479	0.54719
890	1714.407	0.54676
891	1716.336	0.54604
892	1718.264	0.54600
893	1720.193	0.54641
894	1722.121	0.54641
895	1724.050	0.54633
896	1725.978	0.54657
897	1727.906	0.54668
898	1729.835	0.54663
899	1731.763	0.54641
900	1733.692	0.54613
901	1735.620	0.54646
902	1737.549	0.54720
903	1739.477	0.54746
904	1741.406	0.54765
905	1743.334	0.54820
906	1745.263	0.54861
907	1747.191	0.54876
908	1749.120	0.54889
909	1751.048	0.54902
910	1752.977	0.54932
911	1754.905	0.54964
912	1756.833	0.54986
913	1758.762	0.54999
914	1760.690	0.55009
915	1762.619	0.55021
916	1764.547	0.55023
917	1766.476	0.55016
918	1768.404	0.55021
919	1770.333	0.54991
920	1772.261	0.54961
921	1774.190	0.54984
922	1776.118	0.55020
923	1778.047	0.55028
924	1779.975	0.55023
925	1781.904	0.55022
926	1783.832	0.55034
927	1785.760	0.55053
928	1787.689	0.55058
929	1789.617	0.55025
930	1791.546	0.54997
931	1793.474	0.55015
932	1795.403	0.55042
933	1797.331	0.55048
934	1799.260	0.55054
935	1801.188	0.55051
936	1803.117	0.55056
937	1805.045	0.55073
938	1806.974	0.55065
939	1808.902	0.55046
940	1810.831	0.55061
941	1812.759	0.55067
942	1814.688	0.55046
943	1816.616	0.55055
944	1818.544	0.55084
945	1820.473	0.55080
946	1822.401	0.55061
947	1824.330	0.55047
948	1826.258	0.55034
949	1828.187	0.55046
950	1830.115	0.55055
951	1832.044	0.55047
952	1833.972	0.55051
953	1835.901	0.55075
954	1837.829	0.55091
955	1839.758	0.55099
956	1841.686	0.55083
957	1843.615	0.55051
958	1845.543	0.55053
959	1847.471	0.55102
960	1849.400	0.55127
961	1851.328	0.55106
962	1853.257	0.55092
963	1855.185	0.55089
964	1857.114	0.55082
965	1859.042	0.55103
966	1860.971	0.55130
967	1862.899	0.55138
968	1864.828	0.55142
969	1866.756	0.55136
970	1868.685	0.55110
971	1870.613	0.55105
972	1872.542	0.55138
973	1874.470	0.55164
974	1876.398	0.55160
975	1878.327	0.55163
976	1880.255	0.55161
977	1882.184	0.55133
978	1884.112	0.55136
979	1886.041	0.55150
980	1887.969	0.55136
981	1889.898	0.55144
982	1891.826	0.55174
983	1893.755	0.55178
984	1895.683	0.55172
985	1897.612	0.55177
986	1899.540	0.55177
987	1901.469	0.55189
988	1903.397	0.55207
989	1905.325	0.55186
990	1907.254	0.55167
991	1909.182	0.55186
992	1911.111	0.55190
993	1913.039	0.55169
994	1914.968	0.55185
995	1916.896	0.55212
996	1918.825	0.55216
997	1920.753	0.55218
998	1922.682	0.55223
999	1924.610	0.55209
1000	1926.539	0.55204
1001	1928.467	0.55223
1002	1930.396	0.55224
1003	1932.324	0.55202
1004	1934.252	0.55208
1005	1936.181	0.55227

1006	1938.109	0.55239
1007	1940.038	0.55255
1008	1941.966	0.55248
1009	1943.895	0.55224
1010	1945.823	0.55229
1011	1947.752	0.55236
1012	1949.680	0.55225
1013	1951.609	0.55227
1014	1953.537	0.55244
1015	1955.466	0.55259
1016	1957.394	0.55271
1017	1959.323	0.55273
1018	1961.251	0.55259
1019	1963.179	0.55255
1020	1965.108	0.55272
1021	1967.036	0.55270
1022	1968.965	0.55267
1023	1970.893	0.55298
1024	1972.822	0.55303
1025	1974.750	0.55289
1026	1976.679	0.55313
1027	1978.607	0.55321
1028	1980.536	0.55286
1029	1982.464	0.55276
1030	1984.393	0.55284
1031	1986.321	0.55281
1032	1988.250	0.55304
1033	1990.178	0.55328
1034	1992.106	0.55311
1035	1994.035	0.55292
1036	1995.963	0.55309
1037	1997.892	0.55335
1038	1999.820	0.55342
1039	2001.749	0.55334
1040	2003.677	0.55318
1041	2005.606	0.55320
1042	2007.534	0.55343
1043	2009.463	0.55348
1044	2011.391	0.55336
1045	2013.320	0.55336
1046	2015.248	0.55337
1047	2017.177	0.55334
1048	2019.105	0.55343
1049	2021.033	0.55349
1050	2022.962	0.55330
1051	2024.890	0.55321
1052	2026.819	0.55337
1053	2028.747	0.55350
1054	2030.676	0.55359
1055	2032.604	0.55358
1056	2034.533	0.55360
1057	2036.461	0.55358
1058	2038.390	0.55361
1059	2040.318	0.55361
1060	2042.247	0.55345
1061	2044.175	0.55354
1062	2046.104	0.55377
1063	2048.032	0.55372
1064	2049.960	0.55369
1065	2051.889	0.55379
1066	2053.817	0.55367
1067	2055.746	0.55363
1068	2057.674	0.55374
1069	2059.603	0.55365
1070	2061.531	0.55353
1071	2063.460	0.55348
1072	2065.388	0.55351
1073	2067.317	0.55379
1074	2069.245	0.55418
1075	2071.174	0.55420
1076	2073.102	0.55388
1077	2075.031	0.55376
1078	2076.959	0.55383
1079	2078.887	0.55374
1080	2080.816	0.55371
1081	2082.744	0.55368
1082	2084.673	0.55358
1083	2086.601	0.55374
1084	2088.530	0.55399
1085	2090.458	0.55393
1086	2092.387	0.55374
1087	2094.315	0.55385
1088	2096.244	0.55406
1089	2098.172	0.55413
1090	2100.101	0.55403
1091	2102.029	0.55385
1092	2103.958	0.55375
1093	2105.886	0.55371
1094	2107.814	0.55366
1095	2109.743	0.55361
1096	2111.671	0.55373
1097	2113.600	0.55400
1098	2115.528	0.55400
1099	2117.457	0.55393
1100	2119.385	0.55393

Section V. RESULTS

```
*****spectrum***** modeld07
     thickness is 0.0603cm

 T(1038.7) = 0.5033        residual std. dev.= 0.152E-03
 T(1107.9) = 0.3939        residual std. dev.= 0.844E-04
 T(1259.8) = 0.5110        residual std. dev.= 0.233E-03

 BASELINE TRANSMITTANCE AT 1107.9 WAVENUMBERS= 0.5057
====>  ALPHA(BASE) = 0.877    ALPHA(PEAK) = 4.51057

 NET ABSORBANCE PEAK HEIGHT = 0.095142

         ********************************
         * OXYGEN CONTENT IS 19.68 PPMA *
         ********************************



*****spectrum***** modeld12
     thickness is 0.0588cm

 T(1040.5) = 0.5221        residual std. dev.= 0.192E-03
 T(1107.8) = 0.4015        residual std. dev.= 0.255E-03
 T(1259.8) = 0.5346        residual std. dev.= 0.242E-03

 BASELINE TRANSMITTANCE AT 1107.8 WAVENUMBERS= 0.5259
====>  ALPHA(BASE) = 0.336    ALPHA(PEAK) = 4.33700

 NET ABSORBANCE PEAK HEIGHT = 0.102129

         ********************************
         * OXYGEN CONTENT IS 21.99 PPMA *
         ********************************



*****spectrum***** modeld26
     thickness is 0.0552cm

 T(1039.3) = 0.5151        residual std. dev.= 0.189E-03
 T(1107.8) = 0.3974        residual std. dev.= 0.146E-03
 T(1259.8) = 0.5241        residual std. dev.= 0.241E-03

 BASELINE TRANSMITTANCE AT 1107.8 WAVENUMBERS= 0.5179
====>  ALPHA(BASE) = 0.593    ALPHA(PEAK) = 4.78783

 NET ABSORBANCE PEAK HEIGHT = 0.100500

         ********************************
         * OXYGEN CONTENT IS 23.21 PPMA *
         ********************************
```

```
*****spectrum***** modeld36
     thickness is 0.0476cm

 T(1039.3) = 0.5264        residual std. dev.= 0.172E-03
 T(1107.9) = 0.4355        residual std. dev.= 0.995E-04
 T(1259.8) = 0.5357        residual std. dev.= 0.207E-03

 BASELINE TRANSMITTANCE AT 1107.9 WAVENUMBERS= 0.5292
====>  ALPHA(BASE) = 0.303    ALPHA(PEAK) = 3.82890

 NET ABSORBANCE PEAK HEIGHT = 0.072947

         ********************************
         * OXYGEN CONTENT IS 19.00 PPMA *
         ********************************



*****spectrum***** modeld37
     thickness is 0.0602cm

 T(1041.2) = 0.5117        residual std. dev.= 0.146E-03
 T(1108.0) = 0.4043        residual std. dev.= 0.168E-03
 T(1259.8) = 0.5224        residual std. dev.= 0.211E-03

 BASELINE TRANSMITTANCE AT 1108.0 WAVENUMBERS= 0.5149
====>  ALPHA(BASE) = 0.624    ALPHA(PEAK) = 4.12879

 NET ABSORBANCE PEAK HEIGHT = 0.091619

         ********************************
         * OXYGEN CONTENT IS 18.87 PPMA *
         ********************************



*****spectrum***** modeld51
     thickness is 0.0596cm

 T(1040.5) = 0.5201        residual std. dev.= 0.164E-03
 T(1107.9) = 0.4033        residual std. dev.= 0.102E-03
 T(1259.8) = 0.5301        residual std. dev.= 0.249E-03

 BASELINE TRANSMITTANCE AT 1107.9 WAVENUMBERS= 0.5232
====>  ALPHA(BASE) = 0.405    ALPHA(PEAK) = 4.20695

 NET ABSORBANCE PEAK HEIGHT = 0.098407

         ********************************
         * OXYGEN CONTENT IS 20.74 PPMA *
         ********************************
```

sample file name	modeld07.dat
thickness (mils)	2.3740157480E+01
tbase	5.0573972500E-01
tpeak	3.9389457650E-01
abase	8.7606295310E-01
apeak	4.5103899343E+00
anet	3.6343269812E+00
uncertainty	1.0735350129E-02
standard deviation	5.2676600221E+02
ppm (6.28)	1.9683573442E+01

sample file name	modeld12.dat
thickness (mils)	2.3149606299E+01
tbase	5.2596866007E-01
tpeak	4.0144105833E-01
abase	3.3447332586E-01
apeak	4.3352775874E+00
anet	4.0008042615E+00
uncertainty	1.5548042721E-02
standard deviation	5.2676600221E+02
ppm (6.28)	2.1985050762E+01

sample file name	modeld26.dat
thickness (mils)	2.1732283465E+01
tbase	5.1786107475E-01
tpeak	3.9734250844E-01
abase	5.9342275614E-01
apeak	4.7850185869E+00
anet	4.1915958308E+00
uncertainty	1.2867481063E-02
standard deviation	5.2676600221E+02
ppm (6.28)	2.3183221817E+01

FEDERAL INFORMATION PROCESSING STANDARD SOFTWARE SUMMARY

01. Summary date			02. Summary prepared by *(Name and Phone)*	03. Summary action
Yr.	Mo.	Day	Aslan Baghdadi 975-2062	New ☒ Replacement ☐ Deletion ☐
88	05	11	05. Software title: Automatic Determination of the Oxygen Content of Silicon Wafers Polished on Both Sides	Previous Internal Software ID
04. Software date				07. Internal Software ID: DSPOX
Yr.	Mo.	Day		
88	05	11		

06. Short title: DSPOX

08. Software type	09. Processing mode	10. Application area — General		Specific
☐ Automated Data System	☐ Interactive	☐ Computer Systems Support/Utility	☐ Management/ Business	
☒ Computer Program	☐ Batch	☐ Scientific/Engineering	☐ Process Control	
☐ Subroutine/Module	☐ Combination	☐ Bibliographic/Textual	☐ Other	

11. Submitting organization and address

Semiconductor Electronics Division
National Institute of Standards & Technology
Gaithersburg, MD 20899

12. Technical contact(s) and phone

Aslan Baghdadi
301/975-2062

13. Narrative

These programs implement an ASTM Standard Test Method for the Determination of the Oxygen Content of Silicon Wafers That are Polished on Both Sides. The data are initially obtained on a computer-assisted infrared spectrometer.

14. Keywords

Infrared, IR, oxygen, silicon spectrum

15. Computer manuf'r and model	16. Computer operating system	17. Programing language(s)	18. Number of source program statements
1. VAX 11/785 2. IBM AT	1. VMS 4.7 2. DOS 3.2	1. FORTRAN 2. TURBO PASCAL	
19. Computer memory requirements	**20. Tape drives**	**21. Disk/Drum units**	**22. Terminals**

23. Other operational requirements

24. Software availability			25. Documentation availability		
Available ☒	Limited ☐	In-house only ☐	Available ☒	Inadequate ☐	In-house only ☐

26. FOR SUBMITTING ORGANIZATION USE

Standard Form 185
1974 July
U.S. Dept. of Commerce—NBS
(FIPS. Pub. 30)

U.S. Department of Commerce
National Institute of Standards and Technology
(formerly National Bureau of Standards)
Gaithersburg, MD 20899

Stimulating America's Progress
1913-1988